รายงานโครงการวิจัย

วิจัยและพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

Research and Development on Automatic Controller for Irrigation Systems in Durian using Evaporation pan (Epan).

กรมวิชาการเกษตร

ชื่อหัวหน้าโครงการวิจัย

นายวิโรจน์ โหราศาสตร์

Wirot Horasart

ปี พ.ศ. 2564

รายงานโครงการวิจัย

วิจัยและพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

Research and Development on Automatic Controller for Irrigation Systems in Durian using Evaporation pan (Epan).

กรมวิชาการเกษตร

ชื่อหัวหน้าโครงการวิจัย

นายวิโรจน์ โหราศาสตร์

Wirot Horasart

ปี พ.ศ. 2564

คำปรารภ (Foreword หรือ Preface)

โครงการวิจัยและพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าการระเหยน้ำจากถาดวัดการระเหย (Epan) ดำเนินงานในปีงบประมาณ 2563-2564 โดยมีวัตถุประสงค์เพื่อพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าการระเหยน้ำจากถาดวัดการระเหย (Epan) ซึ่งจะช่วยให้เกษตรกรสามารถกำหนดปริมาณการให้น้ำทุเรียนอย่างมีเหตุผล ซึ่งจะทำให้มีการใช้น้ำในการผลิตทุเรียนอย่างมีเหตุผลและมีประสิทธิภาพ

คณะผู้วิจัยหวังเป็นอย่างยิ่งว่า งานวิจัยที่สำเร็จลุล่วงนี้จะมีผู้นำไปใช้ประโยชน์ ซึ่งคณะผู้วิจัยมีความยินดีที่จะถ่ายทอดเทคโนโลยีและองค์ความรู้ที่ได้จากโครงการวิจัยนี้ให้แก่ผู้ที่สนใจเพื่อให้บรรลุวัตถุประสงค์ที่คณะผู้วิจัยได้ตั้งเป้าหมายไว้

.....

(นายวิโรจน์ โหราศาสตร์)

หัวหน้าโครงการวิจัย

กรมวิชาการเกษตร

สารบัญ

	หน้า
กิตติกรรมประกาศ	1
ผู้วิจัย	2
คำอธิบายสัญลักษณ์และคำย่อ	3
บทนำ	4
บทคัดย่อ	6
1. วิจัยและพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)	8
บทสรุปและข้อเสนอแนะ	30
บรรณานุกรม	31
ภาคผนวก ก ผลทดสอบการทำงานระบบควบคุมอัตโนมัติและข้อมูลจำนวนผลผลิตทุเรียน	32
ภาคผนวก ข โปรแกรมระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)	37

กรมวิชาการเกษตร

กิตติกรรมประกาศ

ขอขอบคุณ ศูนย์วิจัยพัฒนาการเกษตรจันทบุรี อ.มะขาม จ.จันทบุรี ที่ให้การสนับสนุนและอนุเคราะห์ สถานที่แปลงทดลอง เครื่องมืออุปกรณ์ เจ้าหน้าที่เก็บข้อมูลและดูแลแปลงต่างๆ เป็นอย่างดี

ขอขอบคุณ ศูนย์วิจัยพืชสวนจันทบุรี จ.จันทบุรี ที่ให้การสนับสนุนและอนุเคราะห์ เจ้าหน้าที่เก็บข้อมูลผลผลิตและดูแลคำแนะนำต่างๆ

ขอขอบคุณ กลุ่มวิจัยปฐพีวิทยา กองวิจัยพัฒนาปัจจัยการผลิตทางการเกษตร ที่ให้การสนับสนุนและอนุเคราะห์ เครื่องมืออุปกรณ์ในการตรวจวัดความชื้นดิน

ขอขอบคุณกลุ่มซ่อมบำรุงรักษา สถาบันวิจัยเกษตรวิศวกรรม ที่ให้การสนับสนุนและอนุเคราะห์ เครื่องมืออุปกรณ์ เจ้าหน้าที่ เพื่อดำเนินการติดตั้งระบบไฟฟ้าต่างๆ

และขอขอบคุณเจ้าหน้าที่ของกลุ่มพัฒนาพื้นที่เกษตร ที่ช่วยในการสร้างแปลงแบบระบบระบายน้ำใต้ดิน รวมถึงการทดสอบและเก็บข้อมูลผลการทดลองจนแล้วเสร็จ

กรมวิชาการเกษตร

ผู้วิจัย

วิโรจน์ โหราศาสตร์
Wirot Horasart

พิมพ์ชนก ทัพภูมิ
Pimchanok Tuppum

บดินทร์ ณ.จินดา
Bodin Na Jinda

ชมภู จันที
Chompoo Juntee

เพ็ญจันทร์ วิจิตร
Phenchan Whijitara

อุทัย ธานี
Uthai Thani

พักตร์วิภา สุทธิวารี
Pakwipa Suttiwaree

กรมวิชาการเกษตร

คำอธิบายสัญลักษณ์และคำย่อ

ETo = ปริมาณการใช้น้ำของพืชอ้างอิง

ETc = ความต้องการใช้น้ำของพืช

Kp = สัมประสิทธิ์ถาดวัดการระเหยสำหรับถาดวัดแบบเอ

Kc = ค่าสัมประสิทธิ์การใช้น้ำของพืช

Epan = ปริมาณการระเหยน้ำจากถาดวัดการระเหยแบบเอ

mm. = มิลลิเมตร

m. = เมตร

กรมวิชาการเกษตร

บทนำ

ทุเรียนเป็นไม้ผลที่มีความสำคัญทางเศรษฐกิจของประเทศไทยชนิดหนึ่ง เป็นพืชที่ทำรายได้ให้แก่เกษตรกรค่อนข้างสูงเมื่อเปรียบเทียบกับพืชชนิดอื่นๆ ในปี 2560 มีมูลค่า 44,235 ล้านบาท (สำนักวิจัยเศรษฐกิจการเกษตร, 2561) และมีแหล่งปลูกที่สำคัญและมีชื่อเสียงในแถบจังหวัดทางภาคตะวันออก เช่น จ.จันทบุรี จ.ระยอง และ จ.ตราด การปลูกทุเรียนเชิงการค้าในปัจจุบันนั้นจำเป็นต้องมีการให้น้ำเพื่อให้ได้ผลผลิตที่มีคุณภาพ ซึ่งระบบการให้น้ำโดยทั่วไปของสวนทุเรียนเป็นแบบมินิสปริงเกอร์ การปลูกทุเรียนให้ได้คุณภาพดีนั้นเกษตรกรต้องให้น้ำเหมาะสมกับช่วงเวลาและอายุของทุเรียน เช่น มีการหยุดให้น้ำเพื่อกระตุ้นให้ออกดอก หรือในช่วงระยะการเจริญเติบโตของผล ระยะ 8-12 สัปดาห์หลังดอกบานจะต้องมีการให้น้ำเพื่อให้การพัฒนาผลเป็นไปอย่างสมบูรณ์ (กรมวิชาการเกษตร, 2547) และการให้น้ำนั้นต้องเพียงพอถ้าน้อยเกินไปจะทำให้ผลทุเรียนมีรูปทรงบิดเบี้ยวและมีขนาดเล็ก ถ้ามีการให้น้ำมากเกินไปจนท่วมขังโคนต้นจะเป็นปัจจัยที่ทำให้เกิดต่างได้ง่ายๆ เช่น โรครากเน่าโคนเน่า ในปัจจุบันเทคโนโลยีมีความเจริญก้าวหน้ามากขึ้นและพัฒนานำไปใช้ทุกภาคส่วนรวมทั้งภาคการเกษตรด้วย เช่น รถแทรกเตอร์ไร้คนขับและระบบควบคุมการให้น้ำพืช เป็นต้น ระบบควบคุมการให้น้ำพืชที่จำหน่ายในท้องตลาดมีหลายแบบ เช่น แบบตั้งเวลา (Irrigation Timers) ระบบนี้สามารถตั้งเวลาโดยตั้งเวลาได้ตั้งแต่ 1 วินาที ถึง 12 ชั่วโมง หรือระบบสั่งการ เปิด - ปิด ปั๊มน้ำผ่าน Application ในโทรศัพท์มือถือ ซึ่งระบบควบคุมดังที่กล่าวมาไม่สามารถคำนวณหาความต้องการน้ำของพืชได้ เกษตรกรจะเป็นผู้กำหนดระยะเวลาการให้น้ำเองตามความเชื่อหรือประสบการณ์ของเกษตรกร แต่ในทางวิชาการหากต้องการคำนวณหาความต้องการใช้น้ำของพืชนั้นมีหลายวิธี แต่วิธีที่ได้รับการยอมรับและนิยมใช้ในงานเกษตรชลประทาน (ธีระพล, 2549) คือการหาปริมาณการใช้น้ำของพืชโดยอาศัยข้อมูลภูมิอากาศหรือเรียกอีกอย่างหนึ่งว่าการหาปริมาณการใช้น้ำของพืชอ้างอิง (Reference Crop Evapotranspiration; ETo) โดยที่สูตรของ Modified Penman และ Pan Method ได้รับความนิยมมากที่สุด แต่สมการของ Modified Penman ต้องใช้ข้อมูลภูมิอากาศเป็นจำนวนมากซึ่งการจะพัฒนาระบบควบคุมอัตโนมัติเป็นเรื่องยากเพราะต้องใช้เซนเซอร์วัดข้อมูลและบันทึกข้อมูลเป็นจำนวนมาก ที่สำคัญเซนเซอร์ที่ใช้วัดไม่มีขายในท้องตลาดทั่วไป ส่วนสูตรของ Pan Method นั้นใช้ค่าปริมาณการระเหยน้ำจากถาดวัดระเหย (Epan) แบบเอ ซึ่งใช้ค่าที่ไม่ซับซ้อนมาก ผลลัพธ์ที่ออกมาเป็นไปในทางเดียวกันกับสูตร Modified Penman การจะพัฒนาระบบควบคุมอัตโนมัติเป็นเรื่องง่ายกว่า เพราะใช้เซนเซอร์วัดข้อมูลและบันทึกข้อมูลจำนวนน้อยกว่า ที่สำคัญเซนเซอร์ที่ใช้วัดมีขายทั่วไปในท้องตลาดและปัจจุบันมีบอร์ดไมโครคอนโทรลเลอร์สำเร็จรูป ที่รวมเอาตัวไมโครคอนโทรลเลอร์ (Microcontroller) และอุปกรณ์อื่นๆ ที่จำเป็นมาไว้ในบอร์ดเดียวกันทำให้ง่ายต่อการนำไปพัฒนาเป็นอุปกรณ์ควบคุมต่างๆ เช่น เครื่องให้ปุ๋ยตามค่าวิเคราะห์ดินแบบแยกถังปุ๋ยสำหรับอ้อย (ขนิษฐ์, 2560)

กรมวิชาการเกษตร

คณะผู้วิจัยจึงมีแนวความคิดที่จะใช้บอร์ดไมโครคอนโทรลเลอร์สำเร็จรูปนี้ และพัฒนาโปรแกรมเพื่อให้อ่านค่าการระเหยน้ำจากถาดวัดการระเหย (Epan) ออกมาเป็นสัญญาณดิจิตอล (Digital) ในการคำนวณหาความต้องการน้ำของพืช (ETc) เพื่อให้ได้ปริมาณน้ำหรือเวลาการให้น้ำที่ถูกต้องตามหลักวิชาการและแม่นยำพร้อมทั้งปรับใช้ให้เหมาะสมกับการปลูกทุเรียน ซึ่งจะเป็นการเพิ่มประสิทธิภาพในการผลิตทุเรียนและการใช้น้ำให้ได้ประโยชน์สูงสุด

กรมวิชาการเกษตร

บทคัดย่อ

การพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan) โดยใช้บอร์ดไมโครคอนโทรลเลอร์ Arduino รุ่น MEGA เขียนโปรแกรมด้วย Arduino Software (IDE) มีการปรับใช้ pressure sensor เพื่อใช้วัดค่าระดับน้ำ ระบบควบคุมอัตโนมัติฯ มีลำดับขั้นตอนการทำงานดังนี้ อ่านค่าจากเซนเซอร์วัดระดับน้ำ (pressure sensor) เพื่อคำนวณหาค่าการระเหย Epan รับค่าจากเซนเซอร์วัดความเร็วลม ความชื้นสัมพัทธ์ในอากาศเพื่อเลือกค่า Kp มีการใส่ข้อมูลค่า Kc, รัศมีทรงพุ่ม, อัตราจ่ายน้ำของหัวมินิสปริงเกอร์ คำนวณหาปริมาณการใช้น้ำของทุเรียน ตามสมการ ETc = Kp x Epan x Kc คำนวณเวลาที่ให้น้ำ ตามสมการ T = IR/q เมื่อประมวลผลเสร็จจะแสดงจำนวนเวลาการให้น้ำ (นาที) ส่งสัญญาณเปิดปั๊มน้ำและโซลินอยด์วาล์ว เมื่อทำงานครบเวลาทั้ง 4 วาล์ว ส่งสัญญาณจบทำงาน ทำการติดตั้งระบบควบคุมฯพร้อมทดลองที่แปลงปลูกทุเรียน สำนักวิจัยพัฒนาการเกษตรจันทบุรี จ.จันทบุรี การทดสอบแบ่งเป็น 2 Mode คือ Mode ที่ 1 ทำงานแบบอัตโนมัติ ค่าระดับน้ำที่วัดได้จากเซนเซอร์มีค่าความคลาดเคลื่อน -68.18 – 76.85% เมื่อเทียบกับค่าที่วัดได้จากตะขอ (Hook) Mode ที่ 2 ทำงานแบบ Manual โดยป้อนค่าการระเหย Epan เปรียบเทียบเวลาการให้น้ำที่คำนวณโดยระบบควบคุมฯ กับเวลาที่คำนวณได้จากสูตร มีความคลาดเคลื่อน 0 - 0.013% ผลการทดสอบช่วงที่ทุเรียนเริ่มออกดอกจะถึงระยะหลังออกดอก 5 เดือน พบว่าปริมาณน้ำแปลงเกษตรกรมีการใช้น้ำที่มากกว่า 120.29% เมื่อเทียบกับแปลงที่ติดตั้งระบบควบคุมอัตโนมัติฯ โดยที่ผลผลิตไม่มีความแตกต่างกันทางสถิติ แต่เมื่อเปรียบเทียบความสมบูรณ์ของต้นทุเรียนหลังการทดลอง พบว่าแปลงที่ติดตั้งระบบควบคุมฯ มีความแตกต่างทางสถิติ โดยต้นทุเรียนมีความสมบูรณ์น้อยกว่า

คำสำคัญ: ระบบควบคุมอัตโนมัติการให้น้ำ, การระเหยน้ำ, ทุเรียน

กรมวิชาการเกษตร

Abstract

Development of automatic irrigation for durian by using Epan with Arduino microcontroller bord Mega model, written program by Arduino software (IDE); equipped with pressure sensor for water measurement. The automatic system starts working from reading water sensor leveling (pressure sensor) for calculation evaporation value (Epan), obtains data from wind speed sensor, air relative humidity for selecting Kp, input Kc, Tree radius, mini springer flow rate, Durian water consumption refer to the equation “Etc = Kp X Epan X Kc “. Calculated watering time using equation “T = IR / q”

After the calculation watering time is showed (min), signaled to open a pump and solenoid valves, signaled back when 4 valves finished their work. The system is set at the Office of Agriculture and Development Region, Chanthaburi at Durian research area. The experiment is designed to two modes, the first mode is autonomous, has discrepancy of water level between 68.18 – 76.85 % compared with the Hook. The second mode in manual, input Epan parameter, compared watering which calculated by the system with the formular; with 0.000 – 0.013 % discrepancy. Test results at flowering period until 5 months after the period show that water usage at farmer treatment is 120.29 % compared to the treatment which equipped with the automatic system. There is no significant in Durian production between the two treatments but the healthiness, farmer treatment significantly shows healthier Durian tress.

Keywords: automatic irrigation system, water evaporation, Durian

กรมวิชาการเกษตร

วิจัยและพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

Research and Development on Automatic Controller for Irrigation Systems in Durian using Evaporation pan (Epan).

วิโรจน์ โหราศาสตร์ Wirot Horasart

พิมพ์ชนก ทัพภูมิ Pimchanok Tuppum

บดินทร์ ณ.จินดา Bodin Na Jinda

ชมภู จันที Chompoo Juntee

เพ็ญจันทร์ วิจิตร Phenchan Whijitara

อุทัย ธานี Uthai Thani

พักตร์วิภา สุทธิวารี Pakwipa Suttiwaree

กรมวิชาการเกษตร

บทคัดย่อ

การพัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan) โดยใช้บอร์ดไมโครคอนโทรลเลอร์ Arduino รุ่น MEGA เขียนโปรแกรมด้วย Arduino Software (IDE) มีการปรับใช้ pressure sensor เพื่อใช้วัดค่าระดับน้ำ ระบบควบคุมอัตโนมัติฯ มีลำดับขั้นตอนการทำงานดังนี้ อ่านค่าจากเซนเซอร์วัดระดับน้ำ (pressure sensor) เพื่อคำนวณหาค่าการระเหย Epan รับค่าจากเซนเซอร์วัดความเร็วลม ความชื้นสัมพัทธ์ในอากาศเพื่อเลือกค่า Kp มีการใส่ข้อมูลค่า Kc, รัศมีทรงพุ่ม, อัตราจ่ายน้ำของหัวมินิสปริงเกอร์ คำนวณหาปริมาณการใช้น้ำของทุเรียน ตามสมการ ETc = Kp x Epan x Kc คำนวณเวลาที่ให้น้ำ ตามสมการ T = IR/q เมื่อประมวลผลเสร็จจะแสดงจำนวนเวลาการให้น้ำ (นาที) ส่งสัญญาณเปิดปั๊มน้ำและโซลินอยด์วาล์ว เมื่อทำงานครบเวลาทั้ง 4 วาล์ว ส่งสัญญาณจบทำงาน ทำการติดตั้งระบบควบคุมฯพร้อมทดลองที่แปลงปลูกทุเรียน สำนักวิจัยพัฒนาการเกษตรจันทบุรี จ.จันทบุรี การทดสอบแบ่งเป็น 2 Mode คือ Mode ที่ 1 ทำงานแบบอัตโนมัติ ค่าระดับน้ำที่วัดได้จากเซนเซอร์มีค่าความคลาดเคลื่อน -68.18 – 76.85% เมื่อเทียบกับค่าที่วัดได้จากตะขอ (Hook) Mode ที่ 2 ทำงานแบบ Manual โดยป้อนค่าการระเหย Epan เปรียบเทียบเวลาการให้น้ำที่คำนวณโดยระบบควบคุมฯ กับเวลาที่คำนวณได้จากสูตร มีความคลาดเคลื่อน 0 - 0.013% ผลการทดสอบช่วงที่ทุเรียนเริ่มออกดอกจะถึงระยะหลังออกดอก 5 เดือน พบว่าปริมาณน้ำแปลงเกษตรกรมีการใช้น้ำที่มากกว่า 120.29% เมื่อเทียบกับแปลงที่ติดตั้งระบบควบคุมอัตโนมัติฯ โดยที่ผลผลิตไม่มีความแตกต่างกันทางสถิติ แต่เมื่อเปรียบเทียบความสมบูรณ์ของต้นทุเรียนหลังการทดลอง พบว่าแปลงที่ติดตั้งระบบควบคุมฯ มีความแตกต่างทางสถิติ โดยต้นทุเรียนมีความสมบูรณ์น้อยกว่า

คำสำคัญ: ระบบควบคุมอัตโนมัติการให้น้ำ, การระเหยน้ำ, ทุเรียน

กรมวิชาการเกษตร

Abstract

Development of automatic irrigation for durian by using Epan with Arduino microcontroller bord Mega model, written program by Arduino software (IDE); equipped with pressure sensor for water measurement. The automatic system starts working from reading water sensor leveling (pressure sensor) for calculation evaporation value (Epan), obtains data from wind speed sensor, air relative humidity for selecting Kp, input Kc, Tree radius, mini springer flow rate, Durian water consumption refer to the equation “Etc = Kp X Epan X Kc “. Calculated watering time using equation “T = IR / q”

After the calculation watering time is showed (min), signaled to open a pump and solenoid valves, signaled back when 4 valves finished their work. The system is set at the Office of Agriculture and Development Region, Chanthaburi at Durian research area. The experiment is designed to two modes, the first mode is autonomous, has discrepancy of water level between 68.18 – 76.85 % compared with the Hook. The second mode in manual, input Epan parameter, compared watering which calculated by the system with the formular; with 0.000 – 0.013 % discrepancy. Test results at flowering period until 5 months after the period show that water usage at farmer treatment is 120.29 % compared to the treatment which equipped with the automatic system. There is no significant in Durian production between the two treatments but the healthiness, farmer treatment significantly shows healthier Durian tress.

Keywords: automatic irrigation system, water evaporation, Durian

กรมวิชาการเกษตร

บทนำ

ทุเรียนเป็นไม้ผลที่มีความสำคัญทางเศรษฐกิจของประเทศไทยชนิดหนึ่ง เป็นพืชที่ทำรายได้ให้แก่เกษตรกรค่อนข้างสูงเมื่อเปรียบเทียบกับพืชชนิดอื่นๆ ในปี 2560 มีมูลค่า 44,235 ล้านบาท (สำนักวิจัยเศรษฐกิจการเกษตร, 2561) และมีแหล่งปลูกที่สำคัญและมีชื่อเสียงในแถบจังหวัดทางภาคตะวันออก เช่น จ.จันทบุรี จ.ระยอง และ จ.ตราด การปลูกทุเรียนเชิงการค้าในปัจจุบันนั้นจำเป็นต้องมีการให้น้ำเพื่อให้ได้ผลผลิตที่มีคุณภาพ ซึ่งระบบการให้น้ำโดยทั่วไปของสวนทุเรียนเป็นแบบมินิสปริงเกอร์ การปลูกทุเรียนให้ได้คุณภาพดีนั้นเกษตรกรต้องให้น้ำเหมาะสมกับช่วงเวลาและอายุของทุเรียน เช่น มีการหยุดให้น้ำเพื่อกระตุ้นให้ออกดอก หรือในช่วงระยะการเจริญเติบโตของผล ระยะ 8-12 สัปดาห์หลังดอกบานจะต้องมีการให้น้ำเพื่อให้การพัฒนาผลเป็นไปอย่างสมบูรณ์ (กรมวิชาการเกษตร, 2547) และการให้น้ำนั้นต้องเพียงพอถ้าน้อยเกินไปจะทำให้ผลทุเรียนมีรูปทรงบิดเบี้ยวและมีขนาดเล็ก ถ้ามีการให้น้ำมากเกินไปจนท่วมขังโคนต้นจะเป็นปัจจัยที่ทำให้เกิดต่างได้ง่ายๆ เช่น โรครากเน่าโคนเน่า ในปัจจุบันเทคโนโลยีมีความเจริญก้าวหน้ามากขึ้นและพัฒนานำไปใช้ทุกภาคส่วนรวมทั้งภาคการเกษตรด้วย เช่น รถแทรกเตอร์ไร้คนขับและระบบควบคุมการให้น้ำพืช เป็นต้น ระบบควบคุมการให้น้ำพืชที่จำหน่ายในท้องตลาดมีหลายแบบ เช่น แบบตั้งเวลา (Irrigation Timers) ระบบนี้สามารถตั้งเวลาโดยตั้งเวลาได้ตั้งแต่ 1 วินาที ถึง 12 ชั่วโมง หรือระบบสั่งการ เปิด - ปิด ปั๊มน้ำผ่าน Application ในโทรศัพท์มือถือ ซึ่งระบบควบคุมดังที่กล่าวมาไม่สามารถคำนวณหาความต้องการน้ำของพืชได้ เกษตรกรจะเป็นผู้กำหนดระยะเวลาการให้น้ำเองตามความเชื่อหรือประสบการณ์ของเกษตรกร แต่ในทางวิชาการหากต้องการคำนวณหาความต้องการใช้น้ำของพืชนั้นมีหลายวิธี แต่วิธีที่ได้รับการยอมรับและนิยมใช้ในงานเกษตรชลประทาน (ธีระพล, 2549) คือการหาปริมาณการใช้น้ำของพืชโดยอาศัยข้อมูลภูมิอากาศหรือเรียกอีกอย่างหนึ่งว่าการหาปริมาณการใช้น้ำของพืชอ้างอิง (Reference Crop Evapotranspiration; ETo) โดยที่สูตรของ Modified Penman และ Pan Method ได้รับความนิยมมากที่สุด แต่สมการของ Modified Penman ต้องใช้ข้อมูลภูมิอากาศเป็นจำนวนมากซึ่งการจะพัฒนาระบบควบคุมอัตโนมัติเป็นเรื่องยากเพราะต้องใช้เซนเซอร์วัดข้อมูลและบันทึกข้อมูลเป็นจำนวนมาก ที่สำคัญเซนเซอร์ที่ใช้วัดไม่มีขายในท้องตลาดทั่วไป ส่วนสูตรของ Pan Method นั้นใช้ค่าปริมาณการระเหยน้ำจากถาดวัดระเหย (Epan) แบบเอ ซึ่งใช้ค่าที่ไม่ซับซ้อนมาก ผลลัพธ์ที่ออกมาเป็นไปในทางเดียวกันกับสูตร Modified Penman การจะพัฒนาระบบควบคุมอัตโนมัติเป็นเรื่องง่ายกว่า เพราะใช้เซนเซอร์วัดข้อมูลและบันทึกข้อมูลจำนวนน้อยกว่า ที่สำคัญเซนเซอร์ที่ใช้วัดมีขายทั่วไปในท้องตลาดและปัจจุบันมีบอร์ดไมโครคอนโทรลเลอร์สำเร็จรูป ที่รวมเอาตัวไมโครคอนโทรลเลอร์ (Microcontroller) และอุปกรณ์อื่นๆ ที่จำเป็นมาไว้ในบอร์ดเดียวกันทำให้ง่ายต่อการนำไปพัฒนาเป็นอุปกรณ์ควบคุมต่างๆ เช่น เครื่องให้ปุ๋ยตามค่าวิเคราะห์ดินแบบแยกถังปุ๋ยสำหรับอ้อย (ขนิษฐ์, 2560)

คณะผู้วิจัยจึงมีแนวความคิดที่จะใช้บอร์ดไมโครคอนโทรลเลอร์สำเร็จรูปนี้ และพัฒนาโปรแกรมเพื่อให้อ่านค่าการระเหยน้ำจากถาดวัดการระเหย (Epan) ออกมาเป็นสัญญาณดิจิตอล

กรมวิชาการเกษตร

(Digital) ในการคำนวณหาความต้องการน้ำของพืช (ETc) เพื่อให้ได้ปริมาณน้ำหรือเวลาการให้น้ำที่ถูกต้องตามหลักวิชาการและแม่นยำพร้อมทั้งปรับใช้ให้เหมาะสมกับการปลูกทุเรียน ซึ่งจะเป็นการเพิ่มประสิทธิภาพในการผลิตทุเรียนและการใช้น้ำให้ได้ประโยชน์สูงสุด

กรมวิชาการเกษตร

การทบทวนวรรณกรรม

กรมวิชาการเกษตร (2547) ได้กล่าวว่า การให้น้ำแก่พืชในปริมาณที่น้อยแต่บ่อยครั้ง จะสามารถรักษาความชื้นในเขตรากพืชให้อยู่ในระดับที่มีความเป็นประโยชน์สูงอยู่เสมอ ได้มากกว่าการให้น้ำปริมาณมากแต่ทิ้งช่วงการให้น้ำห่างกันหลายวัน สำหรับทุเรียนโดยทั่วไปจะให้น้ำทุก 3-5 วัน ขึ้นอยู่กับความสามารถในการอุ้มน้ำของดิน อายุพืช ระยะการเจริญเติบโต เป็นต้น

ธีระพล (2549) การหาปริมาณการใช้น้ำของพืช นั้นสามารถแบ่งออกเป็น 2 แบบ ตามวิธีดำเนินการ คือ แบบที่ 1 การหาปริมาณการใช้น้ำของพืชโดยวิธีการตรวจวัด เช่น การวัดจากถังวัดการใช้น้ำของพืช (Lysimeter Tank) หรือหาจากค่าความชื้นในดินรวมทั้งศึกษาจากแปลงทดลอง ซึ่งวิธีการแบบนี้มีข้อจำกัดและมีค่าใช้จ่ายที่สูงมาก แบบที่ 2 การหาปริมาณการใช้น้ำของพืชโดยอาศัยข้อมูลภูมิอากาศหรือเรียกอีกอย่างหนึ่งว่าการหาปริมาณการใช้น้ำของพืชอ้างอิง (Reference Crop Evapotranspiration; ETo) โดยมีสูตรหรือวิธีการที่ยอมรับและใช้กันอย่างแพร่หลายมีอยู่ด้วยกัน 7 วิธี คือ Modified Penman, Pan Method, Penman Monteith, Blaney Criddle, Thornthwaite, Hargreaves และ Radiation ในงานด้านเกษตรชลประทานจะนิยมใช้ ดังนี้

วิธีที่ 1 Modified Penman มีการคำนวณหาค่า ETo ได้ตามสมการที่ (1)

$$ETo = c[W \cdot Rn + (1 - W) \cdot f(u) \cdot (e_s - e_a)] \quad \text{..... (1)}$$

ETo = ปริมาณการใช้น้ำของพืชอ้างอิง (มม./วัน)

c= ค่าสำหรับใช้ปรับแก้ความคลาดเคลื่อน (Adjustment Factor)

W = แฟกเตอร์ที่อยู่ในเทอมที่เกี่ยวข้องกับรังสีแสงแดด

Rn = รังสีแสงแดดสุทธิ

(1-w) = อิทธิพลของลมและความชื้นในอากาศที่ทราบระดับและอุณหภูมิเฉลี่ย

$f(u)$ = อิทธิพลของกระแสลม

$(e_s - e_a)$ = ผลต่างระหว่างค่าความดันไอน้ำอิ่มตัวเฉลี่ย (e_s) กับความดันไอน้ำที่เป็นจริงเฉลี่ย (e_a)

จากสมการข้างต้นพบว่าการจะพัฒนาระบบควบคุมอัตโนมัติโดยใช้สูตร Modified Penman นั้นเป็นเรื่องยาก เพราะต้องใช้เซนเซอร์เพื่อวัดค่าต่างๆ เป็นจำนวนมากและเซนเซอร์บางชนิดไม่มีขายในท้องตลาดทั่วไป

วิธีที่ 2 Pan Method วิธีนี้ใช้ค่าปริมาณการระเหยน้ำจากถาดวัดระเหย (Epan) แบบเอ ซึ่งใช้ค่าที่ไม่ซับซ้อนมากผลลัพธ์ที่ออกมาเป็นไปในทางเดียวกันกับวิธีที่ 1 Modified Penman โดยมีการคำนวณหาค่า ETo ได้ตามสมการที่ (2)

สมการ ETo = Kp x Epan ..... (2)

ETo = ปริมาณการใช้น้ำของพืชอ้างอิง (มม./วัน)

Kp = สัมประสิทธิ์ถาดวัดการระเหยสำหรับถาดวัดแบบเอ

Epan = ปริมาณการระเหยน้ำจากถาดวัดการระเหยแบบเอ (มม.)

กรมวิชาการเกษตร

การจะพัฒนาระบบควบคุมอัตโนมัติโดยใช้สูตร Pan Method นั้นสามารถทำได้ง่ายกว่า Modified Penman เพราะใช้เซนเซอร์เพื่อวัดค่าต่างๆ จำนวนไม่มากและมีขายในท้องตลาด แต่ของสมการของ Pan Method ยังไม่ได้นำตัวแปรชนิดของพืชมาเกี่ยวข้อง เพื่อให้การคำนวณหาความต้องการใช้น้ำของพืชใกล้เคียงกับความเป็นจริง ดิเรกและคณะ (2545) ได้แนะนำวิธีคำนวณหาปริมาณการใช้น้ำของพืชตามสมการที่ 3 โดยมีการนำค่าสัมประสิทธิ์การใช้น้ำของพืช (Kc) นำมาคำนวณด้วย

สมการ ETc = Kp x Epan x Kc ..... (3)

ETc = ความต้องการใช้น้ำของพืชหรือทุเรียน (มม.)

Kp = สัมประสิทธิ์ถาดวัดการระเหยสำหรับถาดวัดแบบเอ (ตารางที่ 3)

Epan = ปริมาณการระเหยน้ำจากถาดวัดการระเหยแบบเอ (มม.)

Kc = ค่าสัมประสิทธิ์การใช้น้ำของพืช (ตารางที่ 2)

ซึ่งเวลาการให้น้ำสามารถคำนวณจากความสมการดังนี้ (Tayel et. al., 2008)

T = IR/q .....(4) และ IR = ETc x A/Ei

T = เวลาที่ให้น้ำ (ชั่วโมง/รอบ)

q = อัตราการจ่ายน้ำของหัวมินิสปริงเกอร์ (ลิตร/ต้น/ชั่วโมง)

IR = ความต้องการน้ำของพืช (ลิตร/ต้น/รอบ)

ETc = ความต้องการใช้น้ำของพืชหรือทุเรียน (มม.)

A = พื้นที่ทรงพุ่ม (ตารางเมตร)

Ei = ประสิทธิภาพของระบบให้น้ำ = 85% (ดิเรก, 2554) (ตารางที่ 1)

ดิเรกและคณะ (2545) ได้ประยุกต์ใช้ปริมาณการระเหยน้ำจากถาดวัดระเหยแบบ เอ แทน โดยมีค่าประสิทธิภาพการให้น้ำแก่พืชโดยวิธีต่างๆ (ตารางที่ 1) และค่า Kp (ตารางที่ 2)

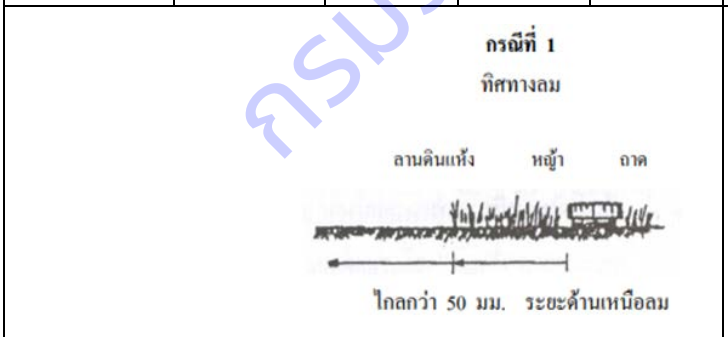
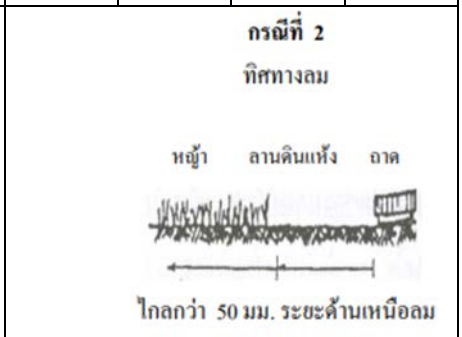
ตารางที่ 1 ประสิทธิภาพการให้น้ำแก่พืชโดยวิธีต่าง ๆ

วิธีการ/ระบบการให้น้ำ	ประสิทธิภาพ (%)
1 การให้น้ำทางผิวดิน	40 - 80
2 การให้น้ำทางใต้ผิวดิน	30 - 50
3 การให้แบบฉีดฝอย	75 - 80
4 การให้น้ำระบบประหยัด	
4.1 สปริงเกอร์	60 - 75
4.2 มินิ/ไมโครสปริงเกอร์	75 - 85
4.3 น้ำหยด	85 - 95

ที่มา : ดิเรก ทองอร่าม และคณะ, 2545

ศูนย์วิทยทรัพยากร

ตารางที่ 2 ค่าสัมประสิทธิ์ของถาดวัดการระเหยสำหรับถาดวัดแบบเอ (Kp)

ความเร็วลมเฉลี่ย (กม./วัน)	กรณีที่ 1 : ถาดล้อมรอบด้วยพืช				กรณีที่ 2 : ถาดวางบนที่ดินว่างเปล่า			
	ระยะด้านเหนือลมที่ปลูกพืช (ม.)	เปอร์เซ็นต์ ความชื้น สัมพัทธ์ ที่ไม่ได้ปลูกพืช			ระยะด้านเหนือลมเฉลี่ย	เปอร์เซ็นต์ ความชื้นสัมพัทธ์		
		20-40	40-70	>70		20-40	40-70	>70
ลมอ่อน <170 กม./วัน	0	0.55	0.65	0.75	0	0.70	0.80	0.85
	10	0.65	0.75	0.85	10	0.60	0.70	0.80
	100	0.70	0.80	0.85	100	0.55	0.65	0.75
	1000	0.75	0.85	0.85	1000	0.50	0.60	0.70
ลมอ่อน - ปานกลาง 170 - 425 กม./วัน	0	0.50	0.60	0.65	0	0.65	0.75	0.80
	10	0.60	0.70	0.75	10	0.55	0.65	0.70
	100	0.65	0.75	0.80	100	0.50	0.60	0.65
	1000	0.70	0.80	0.80	1000	0.45	0.55	0.60
ลมแรง 425 - 700 กม./วัน	0	0.45	0.50	0.60	0	0.60	0.65	0.70
	10	0.55	0.60	0.65	10	0.50	0.55	0.65
	100	0.60	0.65	0.70	100	0.45	0.50	0.60
	1000	0.65	0.70	0.75	1000	0.40	0.45	0.55
ลมแรงมาก >700 กม./วัน	0	0.40	0.45	0.50	0	0.50	0.60	0.65
	10	0.45	0.55	0.60	10	0.45	0.50	0.55
	100	0.50	0.60	0.65	100	0.40	0.45	0.50
	1000	0.55	0.60	0.65	1000	0.35	0.40	0.45
								

ที่มา : ดิเรก ทองอร่าม และคณะ, 2545

กรมวิชาการเกษตร

หิรัญและคณะ (2546) ได้กล่าวว่า ในการคำนวณเพื่อหาค่าความต้องการน้ำของพืช จำเป็นต้องนำข้อมูลสภาพอากาศและข้อมูลพืชมาพิจารณาร่วมกัน แต่เนื่องจากสภาพอากาศมีความแปรปรวนตลอดเวลาและแปรเปลี่ยนไปตามพื้นที่ ดังนั้นหากจะต้องนำข้อมูลสภาพอากาศทุกชนิดที่แปรเปลี่ยนมาบรรจุในสูตรการคิดคำนวณค่าความต้องการน้ำของพืช จะเป็นการไม่สะดวกในทางปฏิบัติจึงได้มีการนำค่าการระเหยของน้ำที่วัดได้จากเครื่องมือวัดอัตราการระเหยของน้ำชนิดถาดแบบ class A (Class A pan) และรวมค่าความแปรปรวนจากสภาพภูมิอากาศและปัจจัยอื่นๆ มาคิดคำนวณร่วมค่าสัมประสิทธิ์การใช้น้ำของพืช Kc (ตารางที่ 3)

ตารางที่ 3 ค่าสัมประสิทธิ์การใช้น้ำของพืช (Kc) ในช่วงต่างๆ ตามการเจริญเติบโตของไม้ผล

ช่วงการเจริญเติบโต	ชนิดของไม้ผล				
	ทุเรียน	เงาะ	มังคุด	ส้ม	ไม้ผลอื่นๆ
1. การพัฒนาการทางกิ่ง ก้าน สาขา	0.60	0.60	0.60	0.65	0.75
2. การชักนำการออกดอก	0.00	0.00/0.60*	0.00	0.00/0.65*	0.00
3. การพัฒนาการของดอก	0.75	0.75	0.75	0.70	0.75
4. การติดผล	0.50	0.75	0.75	0.70	0.75
5. การพัฒนาการของผลอ่อน	0.60	0.80	0.80	0.70	0.75
6. การเจริญเติบโตของผล	0.85	0.85	0.85	0.75	0.80
7. การเริ่มสุกแก่	0.75	0.85	0.85	0.75	0.75

* ในช่วงการชักนำให้ออกดอกของเงาะและส้ม ต้องผ่านช่วงแล้งระยะหนึ่ง จากนั้นจึงเริ่มให้น้ำอย่างสม่ำเสมอ เพื่อกระตุ้นการออกดอก

ที่มา : หิรัญและคณะ, 2546

กรมวิชาการเกษตร

ระเบียบวิธีการวิจัย

- อุปกรณ์

ถาดวัดระเหยแบบเอ (E-pan) , ปั๊มน้ำ 1 HP, วาล์วไฟฟ้า 12V, เครื่องชั่งน้ำหนัก, แปลงปลูกทุเรียนพันธ์หมอนทอง อายุ 5 ปี พื้นที่รวม 2 ไร่

- วิธีการทดลอง

1. ศึกษาวิธีการหาปริมาณความต้องการใช้น้ำของพืชแบบต่างๆ และวิธีการควบคุมระบบให้น้ำพืชอัตโนมัติพร้อมอุปกรณ์ควบคุม เปิด–ปิด โซนให้น้ำ แบบต่างๆ

2. ออกแบบและสร้างอุปกรณ์วัดค่าการระเหยน้ำในถาด Epan พร้อมระบบควบคุมการให้น้ำพืชอัตโนมัติพร้อมทดสอบในห้องปฏิบัติการ

3. ติดตั้งและทดสอบระบบควบคุมการให้น้ำพืชอัตโนมัติโดยใช้ค่าการระเหยน้ำในแปลงปลูกทุเรียนหมอนทอง อายุ 5 ปี ในจังหวัดจันทบุรี ซึ่งแปลงทดสอบนี้มีต้นทุเรียนพื้นที่รวม 2 ไร่ โดยมี 2 กรรมวิธีดังนี้

   1) ปริมาณน้ำตามวิธีของเกษตรกร

   2) ปริมาณน้ำได้จากการคำนวณ 100% ของค่า ETc ที่คำนวณได้

4. เก็บข้อมูลการทำงานของอุปกรณ์ในแปลงทดสอบ เช่น ค่าการระเหยที่ได้จากเครื่องมือเปรียบเทียบกับค่าที่วัดได้จาก Epan มาตรฐาน เพื่อหาความแม่นยำของเครื่องมือ, ปริมาณน้ำที่ได้จากคำนวณ, วัดปริมาณน้ำฝน, เก็บข้อมูลด้านสรีรวิทยา เช่น ความสมบูรณ์ของต้น, ประเมินจำนวนดอก/ต้น, ประเมินจำนวนผล/ต้น เพื่อเปรียบเทียบความแตกต่างทางสถิติ

5. วิเคราะห์ผลการทดลอง

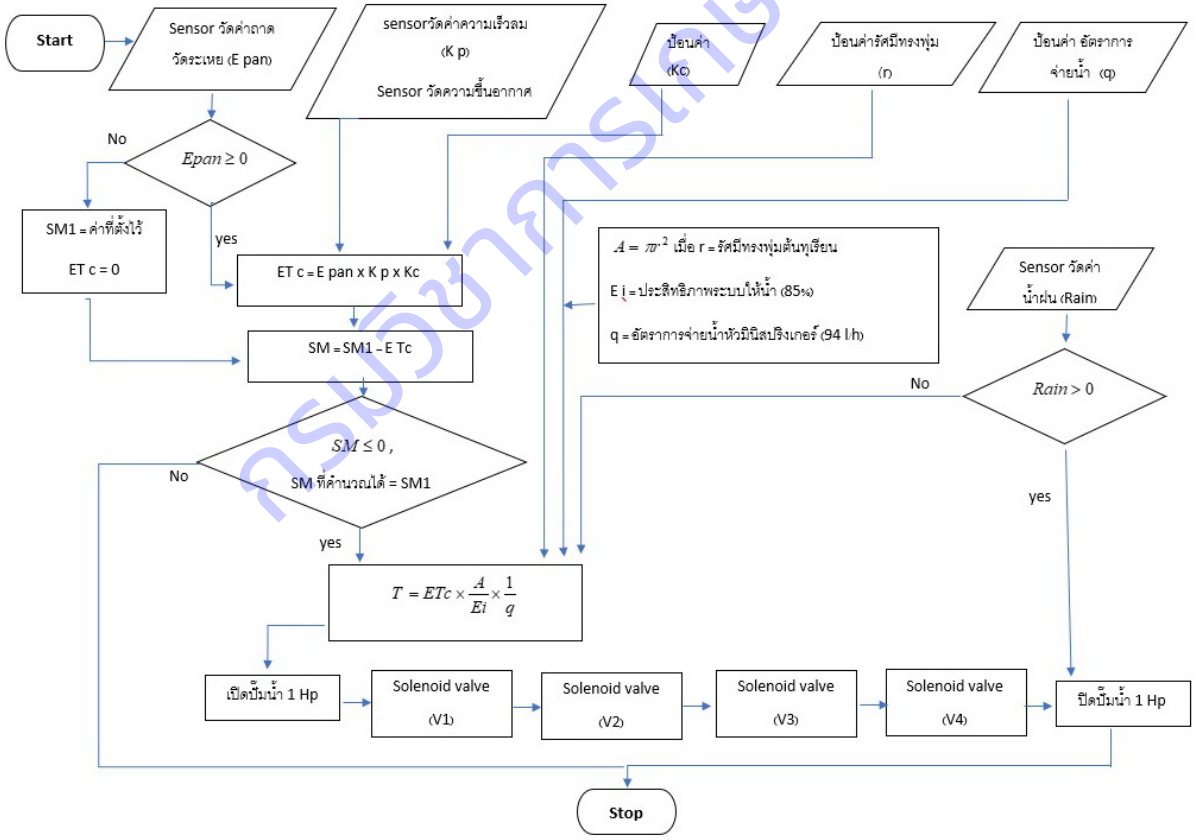
6. สรุปรายงานผลการศึกษาวิจัย จัดทำรายงานผลการดำเนินงาน

กรมวิชาการเกษตร

ผลการวิจัยและอภิปรายผล

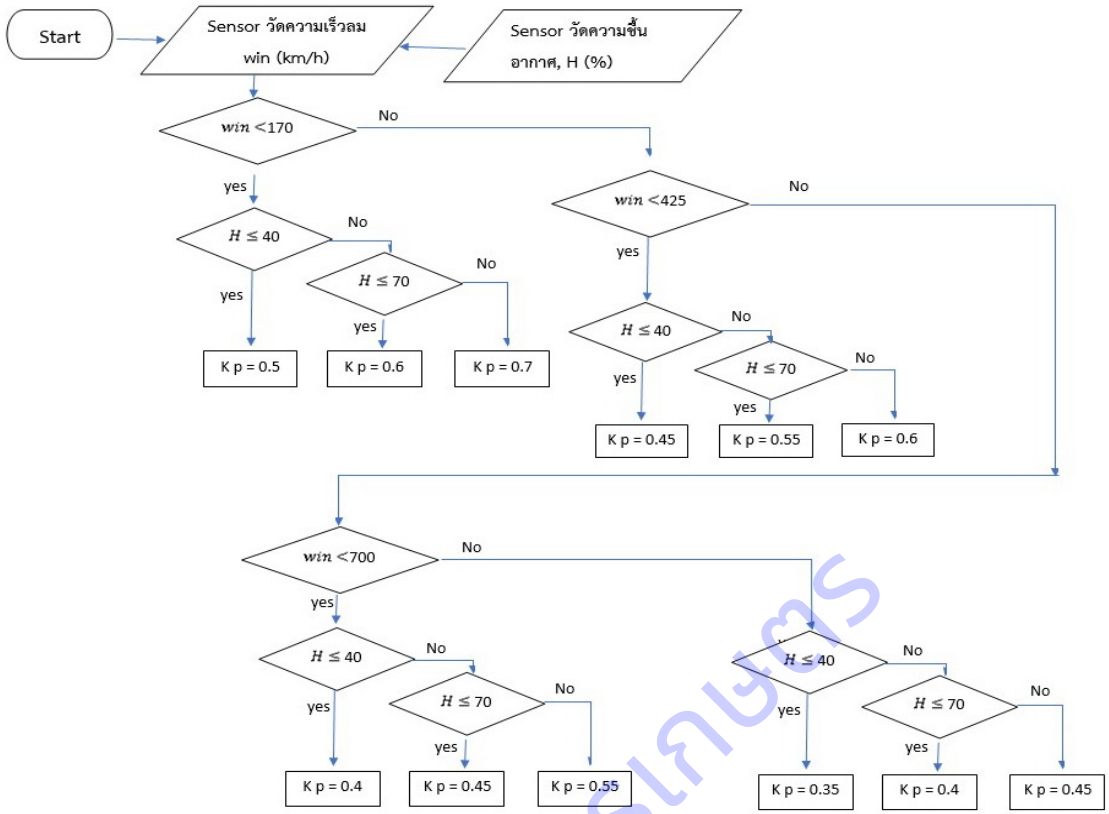
1. การออกแบบระบบควบคุมอัตโนมัติการให้น้ำพืช

งานวิจัยนี้มีแนวความคิดที่จะใช้บอร์ดไมโครคอนโทรลเลอร์สำเร็จรูป และพัฒนาโปรแกรมเพื่อให้อ่านค่าการระเหยน้ำจากถาดวัดการระเหย (Epan) ออกมาเป็นสัญญาณดิจิตอล (Digital) ในการคำนวณหาความต้องการน้ำของพืช (ETc) ตามสมการที่ 3 คำนวณเวลาการให้น้ำตามสมการที่ 4 และมีการเพิ่มค่าสัมประสิทธิ์การใช้น้ำของพืช Kc (ตารางที่ 3) เพื่อให้ได้ปริมาณน้ำที่ถูกต้องตามหลักวิชาการและปรับใช้ให้เหมาะสมกับช่วงอายุของทุเรียน (ช่วงบังคับออกดอก ช่วงติดผล เป็นต้น) ซึ่งมีลำดับการทำงาน Flowchart ดังรูปที่ 1 ระบบควบคุมฯนี้สามารถควบคุมเปิด-ปิด ปั๊มน้ำและแบ่งโซนการให้น้ำได้ประมาณ 4 โซน ติดตั้งเซนเซอร์วัดน้ำฝน (Rain sensor) เพื่อวัดน้ำฝนในกรณีฝนตกในระหว่างที่มีการให้น้ำระบบจะหยุดการทำงานทันที และใช้เซนเซอร์วัดความเร็วลม (wind speed sensor) ติดตั้งด้านข้างของถาดวัดระเหย และเซนเซอร์วัดความชื้นสัมพัทธ์ในอากาศ เพื่อใช้ในการเลือกค่า Kp ตาม Flowchart ดังรูปที่ 2



รูปที่ 1 ลำดับการทำงาน (Flowchart) ของระบบควบคุมอัตโนมัติการให้น้ำพืช

กรมวิชาการเกษตร



รูปที่ 2 ลำดับการทำงาน (Flowchart) เลือกค่า Kp

2. การเขียนโปรแกรมควบคุมไมโครคอนโทรลเลอร์

ชุดควบคุมอัตโนมัติการให้น้ำพืชนี้ใช้บอร์ดไมโครคอนโทรลเลอร์ Arduino จึงใช้โปรแกรม Arduino Software (IDE) ในการเขียนโปรแกรมควบคุมต่างๆ ชุดคำสั่งแบ่งออกเป็น 2 ชุด ดังนี้

ชุดที่ 1 ใช้ควบคุม Arduino รุ่น UNO เพื่อรับค่าจากเครื่องวัดความเร็วลม ซึ่งจะต้องทำงานเก็บค่าความเร็วลมตลอด 24 ชั่วโมง

ชุดที่ 2 ใช้ควบคุม Arduino รุ่น MEGA เพื่อรับค่าจากเซนเซอร์ต่างๆ มาคำนวณหาปริมาณน้ำที่ต้องให้กับพืชและคำนวณเวลาในการเปิด-ปิด ปั๊มน้ำ โดยมีลำดับขั้นตอนการทำงานตาม Flowchart รูปที่ 1 และการเลือกค่า Kp ตาม Flowchart รูปที่ 2

การทำงานของโปรแกรม จะมีเมนูให้เลือกป้อนค่าตัวแปร เช่น ค่าสัมประสิทธิ์การใช้น้ำของพืช (Kc), รัศมีทรงพุ่ม, อัตราการจ่ายน้ำของหัวมินิสปริงเกอร์ เป็นต้น หลังจากตั้งค่าการทำงานต่างๆ แล้ว โปรแกรมจะทำงานตามขั้นตอนใน Flow chart เช่น อ่านค่าจากเซนเซอร์วัดระดับน้ำ (ค่า Epan) รับค่าความเร็วลม (wind speed sensor) จากบอร์ด Arduino รุ่น UNO และเซนเซอร์วัดความความชื้นเพื่อเลือกค่า Kp หลังจากนั้นให้โปรแกรมประมวลผลเพื่อคำนวณเวลาการเปิดปั๊มน้ำและเปิดโซลินอยด์วาล์วที่ควบคุมโซนการให้น้ำ

กรมชลประทาน

3. การทดสอบ sensor ต่างๆ และสร้างระบบควบคุมอัตโนมัติ

ในการสร้างชุดควบคุมอัตโนมัติการให้น้ำพืช จำเป็นต้องใช้เซนเซอร์เพื่อวัดค่าต่างๆ เพื่อนำมาคำนวณหาความต้องการน้ำของพืช (ETc) และระยะเวลาในการเปิด - ปิดปั๊มน้ำตามวัตถุประสงค์ของงานวิจัยฯ ดังนั้น ในการวิจัยนี้ได้มุ่งเน้นที่จะเลือกใช้เซนเซอร์หรืออุปกรณ์ที่มีขายในท้องตลาดทั่วไปมาทดสอบเพื่อใช้งาน ซึ่งได้ผลทดสอบเซนเซอร์วัดระดับน้ำ ดังแสดงในตารางที่ 4

ตารางที่ 4 เปรียบเทียบเซนเซอร์วัดระดับน้ำที่ขายในเชิงพาณิชย์

ลำดับที่	รายการ	รูปภาพ	ผลการทดสอบ	
			ข้อดี	ข้อเสีย
1	เซนเซอร์วัดระดับน้ำแบบResistance		วัดระดับน้ำได้ละเอียดขนาด 0.1 mm.	เมื่อใช้งานต่อเนื่องจะเกิดปฏิกิริยาทางไฟฟ้าเคมี
2	เซนเซอร์วัดระดับน้ำแบบ Capacitive		เมื่อใช้งานต่อเนื่องจะไม่เกิดปฏิกิริยาทางไฟฟ้าเคมี	วัดระดับน้ำได้ละเอียดขนาด 5 mm.

จากตารางที่ 4 พบว่า เซนเซอร์วัดระดับน้ำที่ขายในเชิงพาณิชย์ 2 แบบข้างต้นไม่เหมาะสมที่จะนำมาใช้งานในชุดควบคุมอัตโนมัติการให้น้ำพืช เพื่อแก้ปัญหาการเกิดปฏิกิริยาทางไฟฟ้าเคมีและสามารถวัดระดับน้ำได้ละเอียดในระดับมิลลิเมตร คณะผู้วิจัยจึงได้เลือกนำ pressure Sensor (รูปที่ 3) เซนเซอร์ชนิดนี้มีขายในเชิงพาณิชย์ทั่วไป นำมาปรับปรุงโดยต่อท่อสายยางให้สามารถวัดระดับน้ำ 15 cm ซึ่งมีขนาดความลึกที่เหมาะสม ทำการทดสอบในห้องปฏิบัติการสามารถวัดระดับน้ำที่เปลี่ยนแปลงได้ละเอียดขนาด 1 มิลลิเมตร การทดสอบเปิดชุดให้ทำงานแบบต่อเนื่องเป็นเวลา 3 ชั่วโมง จะไม่ปฏิกิริยาทางเคมี เพราะไม่มีส่วนของไฟฟ้าสัมผัสกับน้ำ และค่าทางไฟฟ้าที่วัดได้จะเปลี่ยนแปลงน้อย ทำการทดสอบโดยการ ปิด-เปิดไฟฟ้า ที่ไม่ต้องทำการ calibration ใหม่ทุกครั้งที่เปิด ทำการทดสอบหาความสัมพันธ์ของค่าไฟฟ้ากับระดับน้ำ (mm) ดังรูปที่ 4 พบว่าที่ระดับ 25 – 55 mm. มีความสัมพันธ์เป็นเส้นตรงตามสมการ $y = 5.0958x + 10812$ มีค่า $R^2 = 0.9499$ ซึ่งจะใช้สมการดังกล่าวในโปรแกรมควบคุมต่อไป

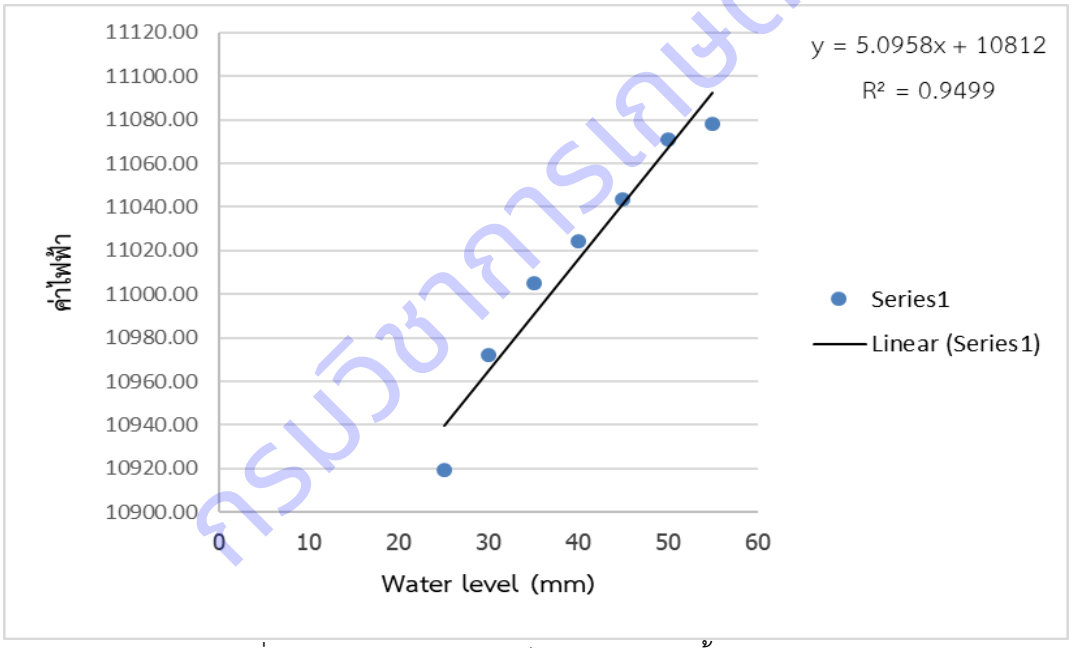
กรมวิชาการเกษตร

(ก) เซนเซอร์ pressure Sensor

(ข) ท่อระดับความลึก 15 cm

(ค) เซนเซอร์วัดระดับน้ำพร้อมแท่น

รูปที่ 3 เซนเซอร์วัดระดับน้ำโดยใช้ pressure Sensor



กรมวิชาการเกษตร

รูปที่ 4 ความสัมพันธ์ของค่าไฟฟ้ากับระดับน้ำ (mm)

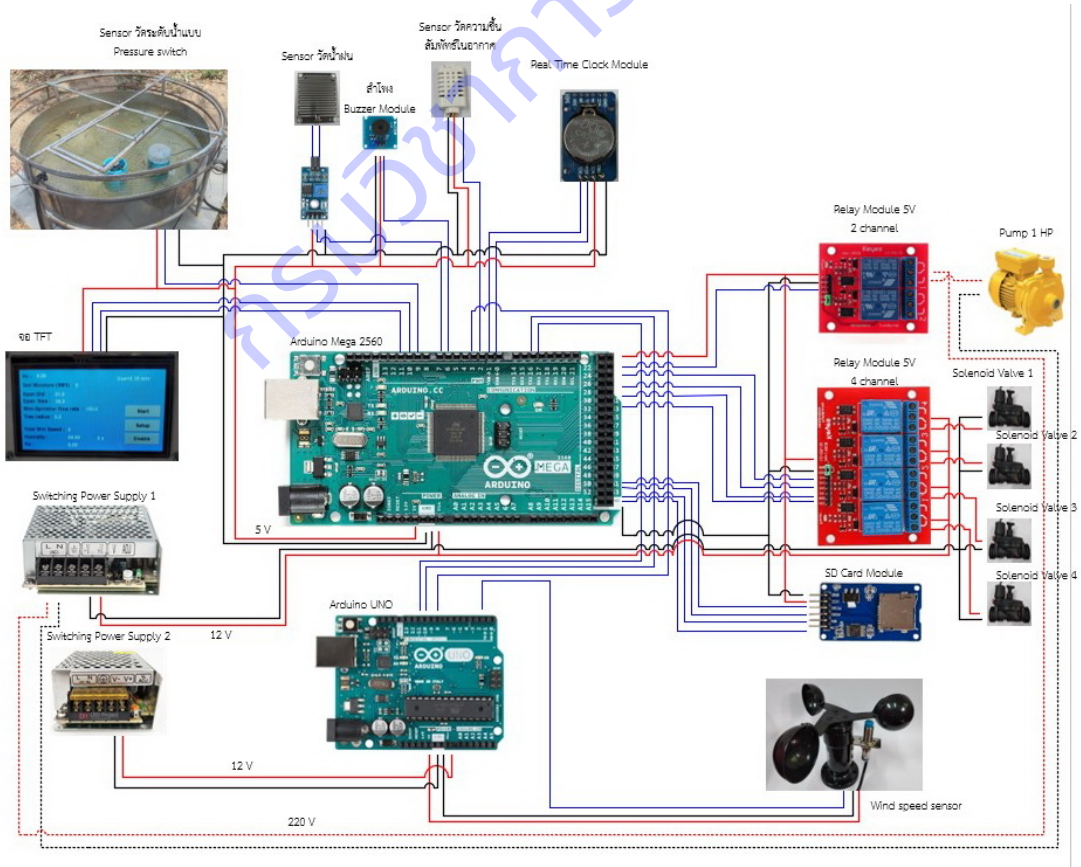
การทดสอบเซนเซอร์วัดความเร็วลม (wind speed sensor) (รูปที่ 5ก) และผลการทดสอบวัดค่าทางไฟฟ้า พบว่าสามารถวัดความเร็วลมได้มีหน่วยเป็น เมตร/วินาที แต่ค่า Kp ตารางที่ 3 ใช้ค่าความเร็วลมรวมต่อวัน เช่น <170 กม./วัน (ลมอ่อน) จึงต้องทำการปรับปรุง โดยใช้ระบบวัดรอบแบบ Proximity Switch DC (รูปที่ 5ข) ในการวัดรอบเพิ่มเติมเพื่อให้สามารถวัดความเร็วลมให้มีหน่วยเป็น กม./วัน ได้

(ก) เซนเซอร์วัดความเร็วรอบ Proximity Switch DC

(ข) เซนเซอร์วัดความเร็วลมแบบนับรอบ

รูปที่ 5 เซนเซอร์วัดความเร็วลม (wind speed sensor) แบบนับรอบ

เมื่อได้ดำเนินการทดสอบและปรับปรุงเซนเซอร์ต่างๆ ให้พร้อมใช้งานในชุดควบคุมอัตโนมัติการให้น้ำพืชตามแผนผังรูปที่ 6 โดยติดตั้งอุปกรณ์ไฟฟ้าในตู้เหล็กกันน้ำและติดตั้งบอร์ดไมโครคอนโทรลเลอร์ Arduino และ อุปกรณ์ต่างๆ ในตู้พลาสติกันน้ำไว้ภายใน การเชื่อมต่อสายไฟฟ้า สายสัญญาณต่างๆ และการดำเนินงานดังรูปที่ 7 โดยราคาชุดควบคุมการให้น้ำพืชอัตโนมัตินี้ ประมาณ 13,000 บาท (ไม่รวมถาดวัดระเหย, ชุดปั๊มน้ำและ Solenoid Valve)



รูปที่ 6 แผนผังวงจรระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

กรมวิชาการเกษตร

(ก) ด้านหน้าชุดควบคุมอัตโนมัติการให้น้ำพืช

(ข) ชุดควบคุมอัตโนมัติการให้น้ำพืชและระบบให้น้ำ

รูปที่ 7 ชุดควบคุมอัตโนมัติการให้น้ำพืชพร้อมทดสอบในห้องปฏิบัติการ

4. ผลการทดสอบในห้องปฏิบัติการ

การทดสอบการทำงานระบบควบคุมอัตโนมัติการให้น้ำพืช โดยเติมน้ำในถาดวัดระเหยให้มีความลึกประมาณ 200 mm. ใช้ตะขอ (Hook) สำหรับวัดระดับน้ำในถาดวัดระเหย มีช่วงการวัดตั้งแต่ 0-100 mm. และความละเอียด 0.1 mm. โดยกำหนดระดับน้ำให้อยู่ในช่วงประมาณ 20-60 mm. ของเครื่องมือวัด ทุกครั้งที่มีการลดระดับน้ำ จะทำการปิดชุดครั้งละประมาณ 30 นาที และเปิดครั้งใหม่ การทดสอบแบ่งเป็น 2 Mode คือ Modeที่ 1 ทำงานอัตโนมัติ Modeที่ 2 ทำงานแบบ Manual (ป้อนค่า Epan เอง) และการทดสอบทั้งสอง Mode ได้ตั้งค่าต่างๆ ดังนี้ Kc = 0.5 (การติดผล), Mini-Sprinkler flow rate = 150 ลิตร/ชั่วโมง, Tree radius = 2.2 เมตร ซึ่งผลการทดสอบได้แสดงดังตารางที่ 8 (ในภาคผนวก ก)

ซึ่งจากผลการทดสอบในตารางที่ 8 พบว่า การวัดระดับน้ำที่วัดจากตะขอ (Hook) ช่วงระดับที่ 22.4 – 57.6 mm. เปรียบเทียบกับระดับน้ำที่วัดได้จากเซนเซอร์มีความแตกต่างกัน เมื่อนำมาคิดความคลาดเคลื่อนของเซนเซอร์วัดระดับน้ำ จะได้ประมาณ 1.75% – 27.03% แต่ที่ระดับ 61.3 mm มีค่าความคลาดเคลื่อนสูงถึง 51.35% อาจเป็นเพราะระดับที่วัดเกินกว่าขอบเขตที่ออกแบบไว้ ซึ่งขั้นตอนการออกแบบและสร้างระบบฯ มีการทดสอบเซนเซอร์วัดระดับน้ำโดยใช้ pressure Sensor ที่ระดับน้ำ 25 – 55 mm. และได้ใช้สมการควบคุมดังที่กล่าว

ผลการทดสอบ Modeที่ 2 ทำงานแบบ Manual โดยป้อนค่าการระเหย Epan ระบบควบคุมอัตโนมัติฯมีการคำนวณเวลาการให้น้ำและเมื่อนำค่าเวลาการให้น้ำที่ได้มาเปรียบเทียบกับเวลาที่คำนวณได้จากสูตร พบว่ามีความแตกต่างกันแต่ไม่มาก และเมื่อคำนวณหาค่าความคลาดเคลื่อนได้

กรมวิชาการเกษตร

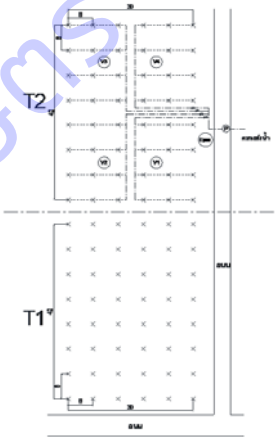
ในช่วง 0 – 0.04% ซึ่งความแตกต่างนี้อาจเป็นผลมาจากการคำนวณที่โปรแกรมคิดเวลาการทำงานเป็นมิลลิวินาทีเมื่อแปลงหน่วยเป็นนาที ทำให้เกิดการปัดทศนิยมของตัวเลขได้

ซึ่งการทำงานโดยรวมของระบบควบคุมอัตโนมัติการให้น้ำพืชสามารถคำนวณปริมาณและเวลาการให้น้ำตามสูตร Epan ได้และมีทำงานได้ตามที่ออกแบบไว้

5. ผลการทดสอบประสิทธิภาพระบบควบคุมอัตโนมัติในแปลงปลูกทุเรียน

ทำการทดสอบในแปลงปลูกทุเรียนอายุ 5 ปี มีระยะการปลูก 6 x 6 เมตร อายุ 5 ปี พื้นที่จำนวน 2.5 ไร่ ณ.ศูนย์วิจัยและพัฒนาการเกษตรจันทบุรี (รูปที่ 8 ก) อ.มะขาม จ.จันทบุรี มีการออกแบบระให้น้ำและจัดแบ่งผังแปลง ดังรูปที่ 8ข เพื่อเปรียบเทียบกรรมวิธี (T1) ปริมาณการให้น้ำแบบเกษตรกร กับกรรมวิธี (T2) ปริมาณน้ำที่ให้ที่คำนวณจากระบบควบคุมการให้น้ำพืชอัตโนมัติโดยใช้ค่าการระเหย (Epan)

(ก) พื้นที่แปลงทดสอบ



(ข) แผนผังแปลงทดสอบพร้อมระบบให้น้ำ

รูปที่ 8 แปลงปลูกทุเรียนที่ใช้ในการทดสอบ

ทำการติดตั้งระบบให้น้ำแบบมินิสปริงเกอร์และระบบควบคุมการให้น้ำพืชอัตโนมัติโดยใช้ค่าการระเหย ในแปลงปลูกทุเรียนพร้อมทดสอบเพื่อหาประสิทธิภาพในการทำงาน ดังรูปที่ 9

กรมวิชาการเกษตร

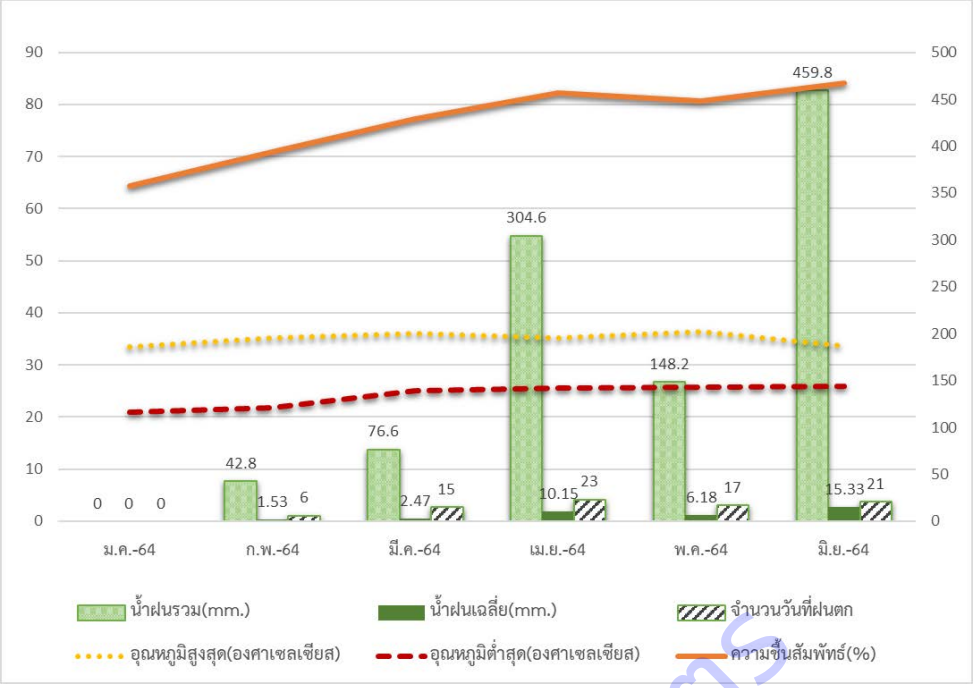
รูปที่ 9 ระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

ในการทดลองนี้ ผู้วิจัยได้ดำเนินได้ควบคุมปัจจัยต่างๆ ให้มีความเท่าเทียมกัน เช่น การดูแลใส่ปุ๋ยต่างๆ ตลอดจนควบคุมจำนวนครั้งในการให้น้ำทุเรียน โดยทั้งสองกรรมวิธีมีแตกต่างกันเฉพาะปริมาณน้ำที่ให้ในแต่ละกรรมวิธีเท่านั้น ปริมาณน้ำในกรรมวิธีที่ 1 เกษตรกรจะให้น้ำครั้งละ 30-60 นาที ซึ่งจะขึ้นอยู่ความพอใจของเกษตรกรเอง ส่วนปริมาณน้ำในกรรมวิธีที่ 2 ระบบควบคุมอัตโนมัติการให้น้ำ จะมีการคำนวณความต้องการใช้น้ำของทุเรียนตามสมการที่ 3 คือ ETc = Kp x Epan x Kc ในการใช้งานระบบควบคุมอัตโนมัติการให้น้ำนี้ ผู้ทำการทดลองกำหนดให้มีการทำงานใน Mode ที่ 2 คือ ทำงานแบบ Manual โดยป้อนค่าการระเหย Epan ซึ่งใน Mode ที่ 2 เวลาการให้น้ำที่ระบบควบคุมคำนวณได้มีความคลาดเคลื่อนเพียง 0-0.013% เมื่อเทียบกับเวลาที่คำนวณได้จากสูตร (ความคลาดเคลื่อนของเวลานี้มาจากการแปลงหน่วยเวลาให้เป็นนาทีและมีการปัดเศษทศนิยม) และข้อมูลบันทึกการทำงานของระบบควบคุมอัตโนมัติในแปลงทดสอบ แสดงในตารางที่ 9 (ภาคผนวก ก) การทำงานใน Mode ที่ 1 ทำงานแบบอัตโนมัตินี้ เซนเซอร์จะวัดค่าการระเหยน้ำในถาด Epan แต่เนื่องจากค่าระดับน้ำที่วัดได้มีค่าความคลาดเคลื่อน -68.18 – 76.85% เมื่อเทียบกับค่าที่วัดได้จากตะขอ (Hook)

ในช่วงระหว่างทำการทดลองได้มีการเก็บข้อมูลทางด้านอุตุนิยมวิทยา เช่น ปริมาณฝน อุณหภูมิ ความชื้นสัมพัทธ์ในอากาศ เป็นต้น ดังแสดงในรูป 10

ทำการเก็บข้อมูลและเปรียบเทียบปริมาณน้ำเฉลี่ยที่ให้ในแต่ละกรรมวิธี ระหว่างวันที่ 15 ม.ค. – 15 มิ.ย. 64 ซึ่งเป็นช่วงที่ทุเรียนเริ่มออกดอกจะถึงระยะหลังออกดอก 5 เดือน (เก็บผลผลิต) ดังแสดงในตารางที่ 5

กรมวิชาการเกษตร



รูปที่ 10 สภาพภูมิอากาศ ณ ศูนย์วิจัยและพัฒนาการเกษตรจันทบุรี 1 ม.ค. 64 – 30 มิ.ย. 64

ตารางที่ 5 ปริมาณน้ำเฉลี่ยที่ให้ในแต่ละกรรมวิธี (ระหว่างวันที่ 15 ม.ค. – 15 มิ.ย. 64)

ช่วงเวลา	จำนวนครั้งที่ให้น้ำ	T1 ปริมาณน้ำเฉลี่ย/ต้น (ลิตร)			T2 ปริมาณน้ำเฉลี่ย/ต้น (ลิตร)		
		ต่อครั้ง	ต่อวัน	ต่อเดือน	ต่อครั้ง	ต่อวัน	ต่อเดือน
15 ม.ค. - 15 ก.พ. 64 หลังออกดอก 1 เดือน	13	183.55	79.54	2386.16	69	29.9	897
16 ก.พ. - 15 มี.ค. 64 หลังออกดอก 2 เดือน	10	168.13	56.04	1681.33	172.11	57.37	1721.1
16 มี.ค. - 15 เม.ย. 64 หลังออกดอก 3 เดือน	8	130.83	34.89	1046.64	183.62	48.97	1468.96
16 เม.ย. - 15 พ.ค. 64 หลังออกดอก 4 เดือน	เว้นการให้น้ำเนื่องจากฝนตก						
16 พ.ค. - 15 มิ.ย. 64 หลังออกดอก 5 เดือน	3	165.04	16.50	495.12	192.01	19.20	576.03
รวม	**34**			**5609.25**			**4663.09**

หมายเหตุ - 15 ม.ค. 64 ทุเรียนเริ่มออกดอก

- ตั้งแต่วันที่ 18 ก.พ 64 กรรมวิธีที่ 2 ต้นทุเรียนแสดงอาการขาดน้ำ จึงให้น้ำ 200% ของค่า Etc

กรมวิชาการเกษตร

จากตารางที่ 5 พบความแตกต่างของปริมาณน้ำเฉลี่ยตั้งแต่ทุเรียนเริ่มออกดอกจะถึงระยะหลังออกดอก 5 เดือน กรรมวิธี T2 ใช้ปริมาณน้ำเฉลี่ย 4663.09 ลิตร/ต้น จะใช้น้ำน้อยกว่าเมื่อเปรียบเทียบกับกรรมวิธี (T1) ของเกษตรกรใช้ปริมาณน้ำเฉลี่ย 5609.25 ลิตร/ต้น ซึ่งใช้น้ำมากกว่าหรือคิดเป็น 120.29 %

แต่ในช่วงทำการทดลองพบว่าในวันที่ 18 กุมภาพันธ์ 2564 ในกรรมวิธีที่ 2 (T2) ต้นทุเรียนเริ่มแสดงอาการขาดน้ำ ผู้วิจัยได้แก้ไขปัญหาโดยเพิ่มปริมาณน้ำเป็น 200% ของค่า ETc เพื่อไม่ให้มีผลกระทบต่อต้นทุเรียนในระยะยาว ซึ่งการให้น้ำโดยใช้ค่า 100% ของค่า ETc ในช่วงแรกอาจไม่เพียงต่อความต้องการน้ำของทุเรียน ทำให้อาการขาดน้ำ ทำให้ผลอ่อนร่วง สาเหตุอีกประการคือ ผู้ทำการทดลองใช้ค่า Kc = 0.6 (ช่วงพัฒนาการทางกิ่ง ก้าน สาขา) ค่าเดียวตลอดช่วงฤดู ไม่ได้ปรับเปลี่ยนค่า Kc ของพืชให้เหมาะสมกับช่วงเวลา เช่น Kc = 0.85 ช่วงการเจริญเติบโตของผล อาจทำให้การคำนวณความต้องการน้ำของทุเรียนผิดพลาดไป

ทำการเก็บข้อมูล ประเมินจำนวนดอก/ต้น ในแต่ละกรรมวิธีตามเทคนิคที่พัฒนาโดยศูนย์วิจัยพืชจันทบุรี โดยการจัดแบ่งกลุ่มกิ่งทุเรียนออกเป็นขนาดต่างๆ ได้แก่ ใหญ่ กลาง และเล็ก ตามขนาดเส้นผ่าศูนย์กลาง ทำการสุ่มนับจำนวนช่อดอกทุเรียนบนกิ่งแต่ละขนาด จัดแบ่งกลุ่มช่อดอกออกเป็นขนาดต่างๆ และสุ่มนับจำนวนดอก/ช่อดอกแต่ละขนาด แล้วจึงนำข้อมูลจำนวนกิ่งแต่ละขนาด จำนวนช่อดอกแต่ละขนาด จำนวนดอก/ช่อมาคำนวณเป็นจำนวนดอกทั้งต้น ตารางที่ 10 (ภาคผนวก ก) และนำค่ามาเปรียบเทียบทางสถิติ ดังนี้ เปรียบเทียบหลังดอกบาน หลังดอกบาน 3 เดือน และน้ำหนักผล แสดงในตารางที่ 6 พบว่าทั้งสองกรรมวิธีไม่มีความแตกต่างกัน

ตารางที่ 6 เปรียบเทียบด้านสรีรวิทยาทางสถิติ

กรรมวิธี	หลังดอกบาน (ดอก/ต้น)	หลังติดผล 3 เดือน (ผล/ต้น)	น้ำหนักผล (กรัม/ผล)
T1	22.00	1.60	3323.86
T2	43.00	4.10	3179.00
t-test	0.198^{ns}	0.335^{ns}	0.130^{ns}

หมายเหตุ ** มีความแตกต่างกันที่ระดับนัยสำคัญ 0.01
* มีความแตกต่างกันที่ระดับนัยสำคัญ 0.05
^{ns} ไม่มีความแตกต่างกันทางสถิติ

จากปริมาณการให้น้ำของกรรมวิธีที่ 1 ในช่วงพัฒนาการของผลหลังออกดอก 2-3 เดือน ปริมาณการให้น้ำประมาณ 34-56 ลิตร/ต้น/วัน และเมื่อเปรียบเทียบกับกรรมวิธีที่ 2 ให้น้ำประมาณ 48-57 ลิตร/ต้น/วัน ซึ่งในช่วงดังกล่าว กรรมวิธีที่ 1 มีการติดผล 1.6 ผล/ต้น และกรรมวิธีที่ 2 ติดผล 4.1 ผล/ต้น ซึ่งการให้น้ำในปริมาณดังกล่าว พบว่า กรรมวิธีที่ 2 ต้นทุเรียนมีสภาพไม่สมบูรณ์เนื่องจาก

กรมวิชาการเกษตร

ให้น้ำไม่เพียงพอ ส่วนหนึ่งอาจมาจากปริมาณการติดผลต่อต้นมากกว่ากรรมวิธีที่ 1 ประมาณ 2 เท่า หากดูการศึกษาปริมาณการให้น้ำทุเรียนของ สุมิตรและคณะ (2561) ในช่วงการพัฒนาการของผลทุเรียน มีการให้น้ำ 100 ลิตร/ต้น/วัน ดังนั้นปริมาณน้ำที่ให้ของกรรมวิธีที่ 2 จึงอาจไม่เพียงพอในช่วงที่ทุเรียนติดผลซึ่งปริมาณการให้น้ำในช่วงดังกล่าวจะต้องสัมพันธ์กับปริมาณผลที่ติด/ต้น

การเก็บข้อมูลผลผลิต แสดงในตารางที่ 11 (ภาคผนวก ก) และแสดงในรูปที่ 11 นำน้ำหนักผลผลิตมาเปรียบเทียบทางสถิติ พบว่า ผลผลิต***ไม่มี***ความแตกต่างกันทางสถิติดังแสดงในตารางที่ 6

(ก) ผลผลิต T1 แปลงเกษตรกร

(ข) ผลผลิต T2 ให้น้ำ100% ของค่า ETc ที่คำนวณได้

รูปที่ 11 ผลทุเรียนหมอนทองทั้งสองกรรมวิธี (เก็บเกี่ยวผลผลิต 15 มิถุนายน 2564)

มีการประเมินสภาพความสมบูรณ์ของต้นทุเรียนก่อนการทดลองในวันที่ 18 พ.ย 2563 ทั้งสองกรรมวิธี (รูปที่ 12) และนำมาเปรียบเทียบความสมบูรณ์ของต้นพบว่าไม่มีความแตกต่างกันทางสถิติ ดังแสดงในตารางที่ 7

การประเมินสภาพความสมบูรณ์ของต้นทุเรียนหลังการเก็บเกี่ยวผลผลิตในวันที่ 31 ส.ค. 2564 และนำมาเปรียบเทียบทางสถิติ พบว่า กรรมวิธีที่ 2 มีความสมบูรณ์ของต้นน้อยกว่ากรรมวิธีที่ 1 ดังแสดงในตารางที่ 7 ซึ่งความสมบูรณ์ต้น ส่วนหนึ่งมาจากปริมาณน้ำที่ให้และปริมาณผลผลิต ซึ่งแม้ในช่วงการพัฒนาผล คือหลังติดผล 2 เดือนจนถึงเก็บเกี่ยว การให้น้ำของกรรมวิธีที่ 2 จะมากกว่า แต่ดูจากจำนวนผลทุเรียนในกรรมวิธีที่ 2 จะมากกว่าประมาณ 2 เท่า ดังนั้นปริมาณน้ำที่ให้จะมากกว่าแต่ก็ยังไม่เพียงพอ จึงทำให้ต้นมีความสมบูรณ์น้อยกว่า

กรมวิชาการเกษตร

(ก) T1 แปลงเกษตรกร ก่อนการทดลอง

(ข) T2 แปลงระบบควบคุมฯ ก่อนการทดลอง

(ค) T1 แปลงเกษตรกร หลังการทดลอง

(ง) T2 แปลงระบบควบคุมฯ หลังการทดลอง

รูปที่ 12 แปลงปลูกทุเรียนก่อนและหลังการทดลอง

กรมวิชาการเกษตร

ตารางที่ 7 คะแนนประเมินความสมบูรณ์ต้นทุเรียน (%) ก่อนและหลังการทดลอง

กรรมวิธี	ก่อนการทดลอง (%)	หลังการทดลอง (%)
T1	82.84	86.53
T2	83.79	74.84
t-test	-1.773^{ns}	-29.607*

หมายเหตุ * มีความแตกต่างกันที่ระดับนัยสำคัญ 0.05
^{ns} ไม่มีความแตกต่างกันทางสถิติ
ก่อนการทดลอง (วันที่ 18 พ.ย 2563)
หลังการทดลอง (วันที่ 31 ส.ค. 2564)

สรุปผลการวิจัยและข้อเสนอแนะ

สถาบันวิจัยเกษตรวิศวกรรมได้พัฒนาระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan) โดยใช้บอร์ดไมโครคอนโทรลเลอร์ Arduino รุ่น MEGA เขียนโปรแกรมด้วย Arduino Software (IDE) เพื่อให้สามารถคำนวณปริมาณความต้องการน้ำของพืชโดยใช้สูตร Pan Method และมีการนำค่าสัมประสิทธิ์การใช้น้ำของพืช (Kc) นำมาร่วมเพื่อให้การคำนวณหาความต้องการใช้น้ำของทุเรียนใกล้เคียงกับความเป็นจริง ตามสมการ ETc = Kp x Epan x Kc และคำนวณเวลาที่ให้น้ำ ตามสมการ T = IR/q และได้ปรับใช้ pressure sensor ให้เป็นเซนเซอร์วัดระดับน้ำ

ทำการติดตั้งระบบควบคุมฯพร้อมทดลองที่แปลงปลูกทุเรียน ณ ศูนย์วิจัยและพัฒนาการเกษตรจันทบุรี จ.จันทบุรี ในระหว่างวันที่ 25 ธ.ค 63 – 24 พ.ค 64 พบว่าการทำงานใน Mode ที่ 1 ทำงานแบบ Auto ค่าระดับน้ำที่วัดได้จากเซนเซอร์เปรียบเทียบกับค่าที่วัดได้จากตะขอ (Hook) มีค่าความคลาดเคลื่อน -68.18 ถึง 76.85% การทำงานใน Mode ที่ 2 คือ ทำงานแบบ Manual โดยป้อนค่าการระเหย Epan ซึ่งการคำนวณเวลาให้น้ำมีความคลาดเคลื่อนน้อยมากเพียง 0 – 0.013% ระบบควบคุมอัตโนมัติการให้น้ำทุเรียนสามารถคำนวณเวลาการให้น้ำได้ซึ่งมีการทำงานได้ตามลำดับขั้นตอนที่ออกแบบไว้

ผลการทดสอบช่วงที่ทุเรียนเริ่มออกดอกจะถึงระยะหลังออกดอก 5 เดือน พบว่าปริมาณน้ำแปลงเกษตรกรมีใช้น้ำที่มากกว่า 120.29% เมื่อเทียบกับแปลงที่ติดตั้งระบบควบคุมฯ ที่มีการทำงานใน Mode ที่ 2 โดยที่ผลผลิตไม่มีความแตกต่างกันทางสถิติ แต่เมื่อเปรียบเทียบความสมบูรณ์ของต้นทุเรียนหลังการทดลอง พบว่าแปลงที่ติดตั้งระบบควบคุมฯ มีความแตกต่างทางสถิติ โดยต้นทุเรียนมีความสมบูรณ์น้อยกว่า

ข้อเสนอแนะ

- ต้องมีการพัฒนาเซนเซอร์ที่ใช้วัดระดับน้ำให้มีความแม่นยำมากยิ่งขึ้น
- ควรมีการทดลองในแปลงซ้ำเพื่อให้ได้ผลการทดลองที่น่าเชื่อถือ

กรมวิชาการเกษตร

บรรณานุกรม

กรมวิชาการเกษตร. 2547. เอกสารวิชาการทุเรียน กรมวิชาการเกษตร กระทรวงเกษตรและสหกรณ์. เอกสารวิชาการ ลำดับที่ 13/2547 : 19-20.

ขนิษฐ์ หว่านณรงค์ อัคคพล เสนาณรงค์ เวียง อากรชี และคณะ. 2560. วิจัยและพัฒนาเครื่องใส่ปุ๋ยตามค่าเคราะห์ดินแบบแยกถังปุ๋ยสำหรับอ้อย. รายงานชุดโครงการวิจัย ปี 2560. กรมวิชาการเกษตร.

ดิเรก ทองอร่าม วิทยา ตั้งก่อสกุล นาวี จิระชีวี และอิทธิสุนทร นันทกิจ. 2545. การออกแบบและเทคโนโลยีการให้น้ำพืช. เจริญรัฐการพิมพ์ กรุงเทพฯ : 57-92.

ธีระพล ตั้งสมบูรณ์. 2549. การใช้น้ำของพืช. เอกสารประกอบการบรรยายหลักสูตรการปรับปรุงระบบการจัดการน้ำด้านเกษตรชลประทาน. กลุ่มงานวิจัยการใช้น้ำชลประทาน. สำนักอุทกวิทยาและบริหารน้ำ.

สำนักวิจัยเศรษฐกิจการเกษตร. 2561. สถิติการเกษตรของประเทศไทย ปี 2560. สำนักงานเศรษฐกิจการเกษตร.

สุมิตร คุณเจตน์; ไพฑูรย์ ศรีนิล; สมบัติ ฝอยทอง; สุคนทิพย์ เถาโมรา; เอื้อน ปิ่นเงิน. 2561. การศึกษาปริมาณความต้องการน้ำและวิธีการให้น้ำที่เหมาะสม สำหรับทุเรียนพันธุ์หมอนทอง. สืบค้นจาก: http://dspace.lib.buu.ac.th/xmlui/handle/1234567890/3948 [ก.พ. 2565].

หิรัญ หิรัญประดิษฐ์ สุขวัฒน์ จันทรปรรณิก เสริมสุข สลักเพ็ชร์. 2546. เทคโนโลยีการผลิตทุเรียน. สำนักพิมพ์มหาวิทยาลัยเกษตรศาสตร์ กรุงเทพฯ. 206 หน้า.

Tayel M.Y., El Gindy A.M. and Abdel-Aziz A.A. (2008). Effect Of Irrigation Systems on: III-Productivity and Quality of Grape Crop. Journal of Applied Sciences Research, 2008, pp. 1722-1729.

กรมวิชาการเกษตร

ภาคผนวก ก

ผลทดสอบการทำงานระบบควบคุมอัตโนมัติและข้อมูลจำนวนผลผลิตทุเรียน

กรมวิชาการเกษตร

ตารางที่ 8 ผลทดสอบการทำงานระบบควบคุมอัตโนมัติการให้น้ำพืชในห้องปฏิบัติการ

ลำดับที่	ระดับน้ำที่วัดจากตะขอ (mm)			ระดับน้ำที่วัดจากเซนเซอร์ (mm)			ความคลาดเคลื่อนของเซนเซอร์วัดระดับน้ำ (%)	ความชื้นสัมพัทธ์ (%)	Kp	Kc	อัตราการจ่ายน้ำ (l/h)	รัศมีทรงพุ่ม (m)	เวลาที่ให้น้ำที่คำนวณโดยระบบควบคุม (min)	เวลาที่ให้น้ำที่คำนวณโดยสูตร (min)	ความคลาดเคลื่อนของเวลาให้น้ำ (%)
	ก่อน	หลัง	ค่า Epan	ก่อน	หลัง	ค่า Epan									
1	61.3	57.6	3.7	57.7	55.9	1.8	51.35	81.1	0.7	0.5	150	2.2	4.51	4.51	-0.10
2	57.6	53.9	3.7	55.9	53.2	2.7	27.03	81.4	0.7	0.5	150	2.2	6.76	6.76	-0.02
3	53.9	48.2	5.7	53.2	47.4	5.8	-1.75	81.6	0.7	0.5	150	2.2	14.52	14.52	-0.01
4	48.2	41.3	6.9	47.4	40.8	6.6	4.35	79.9	0.7	0.5	150	2.2	16.52	16.52	0.00
5	41.3	35.2	6.1	40.8	36.0	4.8	21.31	87.4	0.7	0.5	150	2.2	12.02	12.02	-0.04
6	35.2	29.4	5.8	36.0	29.1	6.9	-18.97	88.4	0.7	0.5	150	2.2	17.27	17.27	0.01
7	29.4	22.4	7.0	29.1	21.0	8.1	-15.71	90.3	0.7	0.5	150	2.2	20.28	20.28	-0.02
8	22.4	15.0	7.4	21.0	12.4	8.6	-16.22	90.2	0.7	0.5	150	2.2	21.53	21.53	-0.01

<u>หมายเหตุ</u> เวลาที่ให้น้ำคำนวณโดยระบบควบคุมฯ เกิดจากการทำงานใน Mode 2

กรมวิชาการเกษตร

ตารางที่ 9 ผลทดสอบการทำงานระบบควบคุมอัตโนมัติการให้น้ำพืชในแปลงทดลอง

ลำดับที่	ค่า Epan (mm)		ความคลาดเคลื่อนของเซนเซอร์วัดระดับน้ำ (%)	ความชื้นสัมพัทธ์ (%)	Kp	Kc	อัตรการจ่ายน้ำ (l/h)	รัศมีทรงพุ่ม (m)	เวลาที่ให้น้ำที่คำนวณโดยเครื่องควบคุม (min)	เวลาที่ให้น้ำที่คำนวณจากสูตร (min)	ความคลาดเคลื่อนของโปรแกรมคำนวณเวลาให้น้ำ %
	ระดับน้ำที่วัดจากตะขอ	ระดับน้ำที่วัดจากเซนเซอร์									
25-ธ.ค.-63	7.7	6.3	18.18	74.2	0.7	0.60	94	2.5	47.66	47.66	0.000
28-ธ.ค.-63	12.6	9.9	21.43	83.3	0.7	0.60	94	2.5	77.99	77.99	-0.001
30-ธ.ค.-63	7.5	6.4	14.67	74.3	0.7	0.60	94	2.5	46.42	46.42	0.005
1-ม.ค.-64	7.2	6.8	5.56	64.8	0.6	0.60	94	2.5	38.2	38.20	-0.003
4-ม.ค.-64	12.3	7.7	37.40	61.9	0.6	0.60	94	2.5	65.26	65.26	-0.006
6-ม.ค.-64	7.7	8.1	-5.19	63.5	0.6	0.60	94	2.5	40.85	40.85	0.004
8-ม.ค.-64	7.3	8.3	-13.70	68.1	0.6	0.60	94	2.5	38.73	38.73	-0.002
11-ม.ค.-64	11.6	10.6	8.62	61.3	0.6	0.60	94	2.5	61.54	61.54	0.004
13-ม.ค.-64	6.7	8.5	-26.87	52.1	0.6	0.60	94	2.5	35.55	35.55	-0.011
15-ม.ค.-64	6.3	4.5	28.57	85.4	0.7	0.60	94	2.5	38.99	38.99	0.012
18-ม.ค.-64	11.7	8.8	24.79	62.9	0.6	0.60	94	2.5	62.07	62.07	0.005
20-ม.ค.-64	6.4	5.3	17.19	56.7	0.6	0.60	94	2.5	33.95	33.95	0.013
22-ม.ค.-64	6.1	1.9	68.85	73.7	0.7	0.60	94	2.5	37.76	37.76	-0.009
25-ม.ค.-64	10.8	2.5	76.85	62.2	0.6	0.60	94	2.5	57.3	57.30	-0.003
27-ม.ค.-64	5.9	3.4	42.37	81.4	0.7	0.60	94	2.5	36.52	36.52	-0.004
29-ม.ค.-64	5.4	5.1	5.56	60.7	0.6	0.60	94	2.5	28.65	28.65	-0.003
1-ก.พ.-64	10.3	5.5	46.60	63.7	0.6	0.60	94	2.5	54.65	54.65	-0.008
3-ก.พ.-64	7.1	6.3	11.27	73.5	0.7	0.60	94	2.5	43.95	43.95	-0.009
5-ก.พ.-64	6.4	7.4	-15.63	57.3	0.6	0.60	94	2.5	33.95	33.95	0.013
8-ก.พ.-64	11.2	8.1	27.68	84.3	0.7	0.60	94	2.5	69.32	69.32	0.005
12-ก.พ.-64	2.9	4.6	-58.62	55.6	0.6	0.60	94	2.5	15.39	15.39	-0.029
15-ก.พ.-64	9.7	10.2	-5.15	81	0.7	0.60	94	2.5	60.04	60.04	-0.001
17-ก.พ.-64	6.6	11.1	-68.18	83.3	0.7	0.60	94	2.5	40.85	40.85	0.004
19-ก.พ.-64	14.8	11.5	22.30	62.3	0.6	0.60	94	2.5	78.52	78.52	0.000
22-ก.พ.-64	23.6	13.7	41.95	77.4	0.7	0.60	94	2.5	146.07	146.07	0.003
24-ก.พ.-64	13.2	13.4	-1.52	78.8	0.7	0.60	94	2.5	81.7	81.70	0.004
26-ก.พ.-64	12.4	9.0	27.42	82	0.7	0.60	94	2.5	76.75	76.75	0.002
1-มี.ค.-64	21.8	16.1	26.15	78.2	0.7	0.60	94	2.5	134.93	134.93	0.003
8-มี.ค.-64	32.2	13.5	58.07	78.8	0.7	0.60	94	2.5	199.31	199.31	-0.002
10-มี.ค.-64	15.6	12.0	23.08	84.4	0.7	0.60	94	2.5	96.56	96.56	-0.002
12-มี.ค.-64	15.2	16.4	-7.89	83	0.7	0.60	94	2.5	94.08	94.08	0.002
15-มี.ค.-64	24.2	18.2	24.79	78.5	0.7	0.60	94	2.5	149.79	149.79	-0.001
19-มี.ค.-64	32.8	25.9	21.04	80.6	0.7	0.60	94	2.5	203.02	203.02	0.000
22-มี.ค.-64	23	23.1	-0.43	90.7	0.7	0.60	94	2.5	142.36	142.36	0.001
24-มี.ค.-64	14.4	19.6	-36.11	55	0.6	0.60	94	2.5	76.4	76.40	-0.003
26-มี.ค.-64	12.4	17.3	-39.52	78.2	0.7	0.60	94	2.5	76.75	76.75	0.002
29-มี.ค.-64	21.8	18.8	13.76	76.3	0.7	0.60	94	2.5	134.93	134.93	0.003
31-มี.ค.-64	14.4	16.1	-11.81	75.4	0.7	0.60	94	2.5	89.13	89.13	0.000
2-เม.ย.-64	14	17.2	-22.86	72.9	0.7	0.60	94	2.5	86.65	86.65	0.005
5-เม.ย.-64	24.2	16.3	32.64	68.1	0.6	0.60	94	2.5	128.39	128.39	0.000
17-พ.ค.-64	40.8	-	-	74.3	0.7	0.60	94	2.5	252.54	252.54	-0.002
21-พ.ค.-64	4	-	-	78.8	0.7	0.60	94	2.5	24.76	24.76	-0.006
24-พ.ค.-64	14.6	-	-	85.1	0.7	0.60	94	2.5	90.37	90.37	-0.002

ลิขสิทธิ์มหาวิทยาลัยเกษตรศาสตร์

ตารางที่ 10 การประเมินจำนวนดอกและ Fruit set

รหัสต้น	ความสมบูรณ์ต้นหลังออกดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	furit set	จำนวนผล/ต้น
T1-1	80	-								0	0
T1-2	80	-								0	0
T1-3	75	20 พ.ย 63	15	15 ม.ค 64	10					16	1
T1-4	75	20 พ.ย 63	15	15 ม.ค 64	20					8	1
T1-5	70	20 พ.ย 63	5	15 ม.ค 64	15					22	1
T1-6	75	-								0	0
T1-7	75	-								0	0
T1-8	75	20 พ.ย 63	5	15 ม.ค 64	15					16	0
T1-9	70	-								0	0
T1-10	75	20 พ.ย 63	10	2 ม.ค 64	10	15 ม.ค 64	15			28	2
T1-11	75	-								0	0
T1-12	75	-								0	0
T1-13	75	20 พ.ย 63	10	2 ม.ค 64	15	15 ม.ค 64	30			41	5
T1-14	75	20 พ.ย 63	15							0	0
T1-15	75	20 พ.ย 63	5	2 ม.ค 64	5	15 ม.ค 64	5			12	0
T1-16	75	-								17	0
T1-17	80	20 พ.ย 63	10	2 ม.ค 64	5					36	4
T1-18	75	20 พ.ย 63	15	2 ม.ค 64	10	15 ม.ค 64	5			24	2
T1-19	75	20 พ.ย 63	20	2 ม.ค 64		15 ม.ค 64	5			4	0
T1-20	75	20 พ.ย 63	5	2 ม.ค 64							

รหัสต้น	ความสมบูรณ์ต้นหลังออกดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	วันออกดอก	% ดอก	furit set	จำนวนผล/ต้น
T2-1	60	2 ม.ค 64	10							24	1
T2-2	60	2 ม.ค 64	5	15 ม.ค 64	5					18	1
T2-3	65	15 พ.ย 63	20	15 ม.ค 64	10					126	19
T2-4	70	15 พ.ย 63	10	2 ม.ค 64	10	15 ม.ค 64	15			142	18
T2-5	70									26	0
T2-6	70									18	0
T2-7	65	20 พ.ย 63	15							7	0
T2-8	65	15 ม.ค 64	5							21	1
T2-9	70	25 ม.ค 64	10							32	1
T2-10	60									0	0
T2-11	65	2 ม.ค 64	5	15 ม.ค 64	5					16	0
T2-12	70	2 ม.ค 64	5	15 ม.ค 64	30					42	0
T2-13	65	20 พ.ย 63	10	2 ม.ค 64	10	10 ม.ค 64	5	15 ม.ค 64	15	159	17
T2-14	65	20 พ.ย 63	10	2 ม.ค 64	10	15 ม.ค 64	5			22	1
T2-15	65	15 พ.ย 63	5	2 ม.ค 64	15	15 ม.ค 64	15	20 ม.ค 64	5	48	3
T2-16	65	20 พ.ย 63	15							0	0
T2-17	65	20 พ.ย 63	5	2 ม.ค 64	5	15 ม.ค 64	5	20 ม.ค 64	5	12	0
T2-18	65									0	0
T2-19	60	10 ม.ค 64	10							42	2
T2-20	70	20 พ.ย 63	20	2 ม.ค 64	25			20 ม.ค 64	40	164	20

กรมวิชาการเกษตร

ตารางที่ 11 การประเมินน้ำหนักผลผลิต

T1/คะแนน.	นน.ผล(กรัม)	ทรงผล	ลักษณะพันธ์	กว้างผล	ยาวผล	เส้นรอบวง	กว้างโคนขั่วผล	กว้างปลายขั่วผล	ยาวขั่วผล	ต่าหนิจากโรค
T1-1	-									
T1-2	-									
T1-3	3140	8	2	18	22	64	2.04	2.62	4.14	3
T1-4	3500	2	4	20	27	63	2.42	2.21	7.76	3
T1-5	3310	8	2	18.5	24	65	2.12	2.51	4.21	3
T1-6	-									
T1-7	-									
T1-8	-									
T1-9	-									
T1-10	1610	1	4	14.5	15	40.5	1.39	1.39	6.5	3
T1-11	-									
T1-12	-									
T1-13	4160	6	3	19.6	30.1	61.6	2.14	2.6	6.7	3
T1-14										
T1-15	-									
T1-16	-									
T1-17	2806	2	4	19.6	21.5	60.5	1.81	2.53	6.1	3
T1-18	4041	0	4	18.5	24	56	2.07	2.53	7.4	3
T1-19	-									
T1-20	-									

T2/คะแนน.	นน.ผล(กรัม)	ทรงผล	ลักษณะพันธ์	กว้างผล	ยาวผล	เส้นรอบวง	กว้างโคนขั่วผล	กว้างปลายขั่วผล	ยาวขั่วผล	ต่าหนิจากโรค
T2-1	2900	2	2	18	24.5	61.0	2.04	2.53	4.18	3
T2-2	3200	2	4	17	27	58.0	2.24	2.43	6.04	3
T2-3	2253	7	2	18.3	21.3	57.7	1.8	2.02	4.32	3
T2-4	2623	8	2	18.3	21	61.5	2.09	2.58	4.36	3
T2-5										3
T2-6	-									
T2-7	-									
T2-8	3420	8	2	19.2	27.5	60	1.86	2.04	4.43	3
T2-9	3380	8	2	18.6	26.8	58	1.92	2.14	5.12	3
T2-10										
T2-11										
T2-12										
T2-13	2347	6	25	16.4	20.2	50.7	1.47	1.79	4.66	3
T2-14	4120	9	2	18.6	24.8	54.8	1.82	2.21	4.46	3
T2-15	3470	9	2	18.5	26.7	59.6	1.97	2.14	4.48	3
T2-16	-									
T2-17	-									
T2-18	-									
T2-19	3640	9	2	18.1	23.1	52.7	1.77	2.08	4.43	3
T2-20	3260	9	2	19.7	22	63.5	1.94	2.21	5.62	3

กรมวิชาการเกษตร

ภาคผนวก ข

โปรแกรมระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

กรมวิชาการเกษตร

โปรแกรมระบบควบคุมอัตโนมัติการให้น้ำทุเรียนโดยใช้ค่าจากถาดวัดระเหย (Epan)

```
#include <EEPROM.h>
int adr_epo_hun = 30;
int adr_epo_ten = 0;
int adr_epo_one = 1;
int adr_epo_p1 = 2;
int adr_epo_p2 = 3;
int adr_flow_hun = 4 ;
int adr_flow_ten = 5 ;
int adr_flow_one = 6 ;
int adr_tree_ten = 7;
int adr_tree_one = 8;
int adr_tree_p1 = 9;
int adr_kc_ten = 20;
int adr_kc_one = 21;
int adr_kc_p1 = 22;
int adr_kc = 10;
int adr_sm1 = 11;
int adr_sm1_begin = 12;
String dater = "";
String tter  = "";
String sme = "";
String Kkc;
//-----------------------------------------
int adr_increment_ten_thousand = 13;
int adr_increment_one_thousand = 14;
int adr_increment_one_hundred = 15;
int adr_increment_one_ten = 16;
int adr_increment_one_one = 17;
int adr_increment_p1 = 18;
int adr_increment_p2 = 19;
int ss = 0;
```

```
int buzzer = 6;
float increment = 0.00;
int settime_begin = 30; //
//-------------------------------------------------------------
#include <Q2HX711.h>
const byte hx711_data_pin = 8;
const byte hx711_clock_pin = 9;
Q2HX711 hx711(hx711_data_pin, hx711_clock_pin);
float WATERValue = 0;
int io;
long WAter_test = 0;
//-------------------------------------------------------------
unsigned long previous = 0;
const long interval2 = 1000;
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
float mm;
#include <SoftwareSerial.h>
#include <Nextion.h>
SoftwareSerial nextion(10, 11);
Nextion myNextion(nextion, 9600);
#define epan A0
#define rain A1
float mins = 0.00;
float Etc;
float sm;
float T;
float Ei = 0.85;
int ts = 0;
```

กรมวิชาการเกษตร

```
char data[6];
String(eepo).toCharArray(data, 6);
//-----------------------------------------------
if ( eepo <= 999 )
{
 int one = data[0] - 48;
 int p1 = data[1] - 48;
 int p2 = data[2] - 48;
 EEPROM.write(adr_epo_hun, 0);
 EEPROM.write(adr_epo_ten, 0);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
if (  eepo >= 1000 && eepo <= 9999 )
{
 int ten = data[0] - 48;
 int one = data[1] - 48;
 int p1 = data[2] - 48;
 int p2 = data[3] - 48;
 EEPROM.write(adr_epo_hun, 0);
 EEPROM.write(adr_epo_ten, ten);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
//-----------------------------------------------------
if (  eepo >= 10000 && eepo <= 99999 )
{
 int hun = data[0] - 48;
 int ten = data[1] - 48;
 int one = data[2] - 48;
 int p1 = data[3] - 48;
 int p2 = data[4] - 48;
 EEPROM.write(adr_epo_hun, hun);
 EEPROM.write(adr_epo_ten, ten);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
 // delay(1000);
 sec = 0;
 steps = 0;
 starts = 0;
 //  EEPROM.write(adr_sec, sec);
 //  EEPROM.write(adr_starts, starts);
 //  EEPROM.write(adr_steps, steps);
}
}
} // end millis
}// end loop
void beep ()
{
 digitalWrite(buzzer, HIGH);
 delay(80);
 digitalWrite(buzzer, LOW);
 delay(80);
}
```

ศรีปทุมการเกษตร

```
  Serial2.println('0');
  digitalWrite(trig, LOW); // Send to reset
  delay(1000);
  digitalWrite(trig, HIGH);
  delay(1000);
  settime_begin = 30;
  enable = 1;
 }
 if ( bstart == 1 )
 {
  if  ( Serial2.available ())
  {
   String wind = Serial2.readString();
   Serial.println("                    start recive : " + String(wind));
   int total_wind1 = wind.toInt();
   Serial.println("                    total_wind  : " + String(total_wind1));
   total_wind = total_wind1 * 0.0005024 ;
   Serial.println("                    total_wind ok  : " + String(total_wind1, 6));
   digitalWrite(buzzer, HIGH);
   delay(3000);
   digitalWrite(buzzer, LOW);
  }
  starts = 1;
  beep ();
  enable = 1;
  myNextion.setComponentText("t999", "Start!!!");
  Serial.println("                    เริ่มทำงาน !!!  ");
  humidity = dht.readHumidity();
  delay(2000);
  Serial.println("                    humidity  : " + String(humidity));
```

```
   mins += 1;
   sec = 0;
   //    EEPROM.write(adr_sec, sec);
   //   EEPROM.write(adr_mins, mins);
  }
  // String ttt = "Time = " + String(mins) + ":" + String(sec) + "  ";
  String ttt = "UT=" + String(tm, 2) + ",CT=" + String(mins) + ":" + String(sec) + "  ";
  myNextion.setComponentText("t999", ttt);
  //  Serial.println("              min4 = " + String(mins));
  //   Serial.println("              sec4 = " + String(sec));
  //    Serial.println("                  Value 4 - ON ");
  digitalWrite(solenoid1, HIGH); // value 1 - ON
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, LOW);
 }
 if ( (mins + sec / 100.00) * 60.00  >= ts && steps == 3 )
 {
  myNextion.setComponentText("t999", "Finish4");
  Serial.println("                  Value 4 - OFF ");
  digitalWrite(solenoid1, HIGH);
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, HIGH);
  delay(1000);
  Serial.println("                Pump - OFF ");
  digitalWrite(pump, HIGH); // pump OFF
  delay(3000);
  epan_old = epan_new;
  int eepo = epan_old * 100;
```

กรมวิชาการเกษตร

```
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm5 == 1 )
{
 sf += "5";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm6 == 1 )
{
 sf += "6";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm7 == 1 )
{
 sf += "7";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm8 == 1 )
{
 sf += "8";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
```

```
}
if ( bm9 == 1 )
{
 sf += "9";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm_point == 1 )
{
 sf += ".";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm_del == 1 )
{
 sf = "";
 flow_rate = sf.toFloat();
 EEPROM.write(adr_flow_hun, 0);
 EEPROM.write(adr_flow_ten, 0);
 EEPROM.write(adr_flow_one, 0);
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
 // if ( bm0 == 0 && bm1 == 0 && bm2 == 0 && bm3 == 0 && bm4 == 0 && bm5 == 0 && bm6 == 0
&& bm7 == 0 && bm8 == 0 && bm9 == 0)
 // {
 int flows = flow_rate; // 10-999
 //Serial.println("                    flows =  " + String(flows));
 char datas[5];
```

กรมวิชาการเกษตร

```
}
if ( bs9 == 1 )
{
 tr += "9";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs_point == 1 )
{
 tr += ".";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs_del == 1 )
{
 tr = "";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
 EEPROM.write(adr_tree_ten, 0);
 EEPROM.write(adr_tree_one, 0);
 EEPROM.write(adr_tree_p1, 0);
}
int trds = tree_radius * 10; // 9.5  , 45.4
char datac[5];
String(trds).toCharArray(datac, 5);
if ( trds <= 99 )
{
 int one = datac[0] - 48;
 int p1 = datac[1] - 48;
 EEPROM.write(adr_tree_ten, 0);
 EEPROM.write(adr_tree_one, one);
 EEPROM.write(adr_tree_p1, p1);
 myNextion.setComponentText("t014", String(tree_radius, 1) );
}
if ( trds >= 100 && trds < 999 )
{
 int ten = datac[0] - 48;
 int one = datac[1] - 48;
 int p1 = datac[2] - 48;
 EEPROM.write(adr_tree_ten, ten);
 EEPROM.write(adr_tree_one, one);
 EEPROM.write(adr_tree_p1, p1);
 myNextion.setComponentText("t014", String(tree_radius, 1) );
}

//================================================================================================== Mini sprikle flow-rate
 int bm0 = myNextion.getComponentValue("bm0");
 int bm1 = myNextion.getComponentValue("bm1");
 int bm2 = myNextion.getComponentValue("bm2");
 int bm3 = myNextion.getComponentValue("bm3");
 int bm4 = myNextion.getComponentValue("bm4");
 int bm5 = myNextion.getComponentValue("bm5");
 int bm6 = myNextion.getComponentValue("bm6");
 int bm7 = myNextion.getComponentValue("bm7");
 int bm8 = myNextion.getComponentValue("bm8");
 int bm9 = myNextion.getComponentValue("bm9");
 int bm_point = myNextion.getComponentValue("bm88");
 int bm_del = myNextion.getComponentValue("bm99");
```

กรมวิชาการเกษตร

```
if ( bs0 == 1 )
{
 tr += "0";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs1 == 1 )
{
 tr += "1";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs2 == 1 )
{
 tr += "2";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs3 == 1 )
{
 tr += "3";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs4 == 1 )
{
 tr += "4";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs5 == 1 )
{
 tr += "5";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs6 == 1 )
{
 tr += "6";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs7 == 1 )
{
 tr += "7";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs8 == 1 )
{
 tr += "8";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
```

กรมวิชาการเกษตร

```
{
  Kkc += "9";
  kc = Kkc.toFloat();
  delay(1000);
  myNextion.setComponentText("kt22", String(kc, 2) );
  beep();
}
if ( bkt_point == 1 )
{
  Kkc += ".";
  kc = Kkc.toFloat();
  delay(1000);
  myNextion.setComponentText("kt22", String(kc, 2) );
  beep();
}
if ( bkt_del == 1 )
{
  Kkc = "";
  kc = Kkc.toFloat();
  delay(1000);
  myNextion.setComponentText("kt22", String(kc, 2) );
  beep();
}
int kcs = kc * 100;
char datak[5];
String(kcs).toCharArray(datak, 5);
if ( kcs <= 99 )
{
  int one = datak[0] - 48;
  int p1 = datak[1] - 48;
  EEPROM.write(adr_kc_ten, 0);
  EEPROM.write(adr_kc_one, one);
  EEPROM.write(adr_kc_p1, p1);
  myNextion.setComponentText("kt22", String(kc, 2) );
}
/*
 if ( kcs >= 100 && kcs <= 999)
 {
 int ten = datak[0] - 48;
 int one = datak[1] - 48;
 int p1 = datak[2] - 48;
 EEPROM.write(adr_kc_ten, ten);
 EEPROM.write(adr_kc_one, one);
 EEPROM.write(adr_kc_p1, p1);
 myNextion.setComponentText("kt22", String(kc, 1) );
 }
*/
//=============================================
Tree Radius
 int bs0 = myNextion.getComponentValue("bs0");
 int bs1 = myNextion.getComponentValue("bs1");
 int bs2 = myNextion.getComponentValue("bs2");
 int bs3 = myNextion.getComponentValue("bs3");
 int bs4 = myNextion.getComponentValue("bs4");
 int bs5 = myNextion.getComponentValue("bs5");
 int bs6 = myNextion.getComponentValue("bs6");
 int bs7 = myNextion.getComponentValue("bs7");
 int bs8 = myNextion.getComponentValue("bs8");
 int bs9 = myNextion.getComponentValue("bs9");
 int bs_point = myNextion.getComponentValue("bs88");
 int bs_del = myNextion.getComponentValue("bs99");
```

กรมวิชาการเกษตร

```
{
 Kkc += "1";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep(); }
if ( bkt2 == 1 )
{
 Kkc += "2";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep(); }
if ( bkt3 == 1 )
{
 Kkc += "3";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt4 == 1 )
{
 Kkc += "4";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt5 == 1 )
{
 Kkc += "5";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt6 == 1 )
{
 Kkc += "6";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt7 == 1 )
{
 Kkc += "7";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt8 == 1 )
{
 Kkc += "8";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt9 == 1 )
```

กรมวิชาการเกษตร

```
  //-------------------------------------------------------------------------------------
  int adc0 = analogRead(sensor);
  Serial.println("                                adc0 = " + String(adc0));
  epan_new = ((adc0 - 640.09) / 5.7089)*-1;
  Serial.println("                                        After  epan_new = " + String(epan_new,
2));
  /*
    for (io = 0; io < 10; io++)
    {
     WATERValue  += hx711.read() / 1000;
    }
    WATERValue  /= 10;
    WAter_test = map(WATERValue, 9530     , 10250   , 0    , 100) ;
    if ( WAter_test <= 0 ) WAter_test = 0;
    else if ( WAter_test >= 100 ) WAter_test = 100;
    Serial.print("Water Lavel: ");
    Serial.print(WAter_test / 100.0);
    Serial.print(" L");
    Serial.print("  ADC  ");
    Serial.println(WATERValue);
    Serial.print("\t\t");
    // X= ระดับน้ำ
    // Y= ค่าที่วัดได้
    epan_new = (WATERValue - 14282) / 6.6069;
//************************************************************** แก้ไขสมการ จุดที่ 2
    //epan_new = 50;
  */
  Serial.println(epan_new, 2);
  myNextion.setComponentText("t103", String(epan_new, 1) );
  Serial2.println('0');
  delay(100);  }
```

```
  if ( benable == 1 )
  {
   beep ();
   enable = 1;
   myNextion.setComponentText("t856", " " );  }
 //============================================================================ Set
KC

//============================================================================
============================================== Set Epan Old
  int bkt0 = myNextion.getComponentValue("kt0");
  int bkt1 = myNextion.getComponentValue("kt1");
  int bkt2 = myNextion.getComponentValue("kt2");
  int bkt3 = myNextion.getComponentValue("kt3");
  int bkt4 = myNextion.getComponentValue("kt4");
  int bkt5 = myNextion.getComponentValue("kt5");
  int bkt6 = myNextion.getComponentValue("kt6");
  int bkt7 = myNextion.getComponentValue("kt7");
  int bkt8 = myNextion.getComponentValue("kt8");
  int bkt9 = myNextion.getComponentValue("kt9");
  int bkt_point = myNextion.getComponentValue("kt88");
  int bkt_del = myNextion.getComponentValue("kt99");
  if ( bkt0 == 1 )
  {
   Kkc += "0";
   kc = Kkc.toFloat();
   delay(1000);
   //-----------------------------------------
   myNextion.setComponentText("kt22", String(kc, 2) );
   beep();
  }
  if ( bkt1 == 1 )
```

ศูนย์วิทยทรัพยากร

```
myNextion.setComponentText("t103", String(epan_new, 1) );
myNextion.setComponentText("t005", String(flow_rate, 1) );
myNextion.setComponentText("t006", String(total_wind) );
myNextion.setComponentText("t765", String(humidity) );
myNextion.setComponentText("t766", String(kp, 2) );
myNextion.setComponentText("t008", String(tree_radius, 1) );
int bstart = myNextion.getComponentValue("b0");
int benable = myNextion.getComponentValue("b1");
Serial.println("                bstart =  " + String(bstart));
Serial.println("                benable =  " + String(benable));
Serial.println();
if (settime_begin <= 0 )
{
 if ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("                recive : " + String(wind));
  int total_wind1 = wind.toInt();
  Serial.println("                total_wind  : " + String(total_wind1));
  total_wind = total_wind1 * 0.0005024 ;
  digitalWrite(buzzer, HIGH);
  delay(3000);
  starts = 1;
  delay(1000);
  digitalWrite(buzzer, LOW);
 }
 myNextion.setComponentText("t999", "Auto-Start");
 Serial.println("                เริ่มทำงาน !!!  ");
 humidity = dht.readHumidity();
 delay(2000);
 Serial.println("                humidity  :  " + String(humidity));
 Serial2.println('0');
 digitalWrite(trig, LOW); // Send to reset
 delay(1000);
 digitalWrite(trig, HIGH);
 delay(1000);
 settime_begin = 30;
 enable = 1;
}
if ( bstart == 1 )
{
 if  ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("                start recive : " + String(wind));
  int total_wind1 = wind.toInt();
  Serial.println("                total_wind  : " + String(total_wind1));
  total_wind = total_wind1 * 0.0005024 ;
  Serial.println("                total_wind ok  : " + String(total_wind1, 6));
  digitalWrite(buzzer, HIGH);
  delay(3000);
  digitalWrite(buzzer, LOW);
 }
 starts = 1;
 beep ();
 enable = 1;
 myNextion.setComponentText("t999", "Start!!!");
 Serial.println("                เริ่มทำงาน !!!  ");
 humidity = dht.readHumidity();
 delay(2000);
 Serial.println("                humidity  :  " + String(humidity));
```

กรมวิชาการเกษตร

```
// steps = EEPROM.read(adr_steps);
// readsensor ();
/*
 if ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("recive : " + String(wind));
  total_wind = wind.toInt();
  Serial.println("                    total_wind  : " + String(total_wind));
  digitalWrite(buzzer, HIGH);
  delay(3000);
  starts = 1;
  delay(1000);
  digitalWrite(buzzer, LOW);
 }
*/
//========================================================= Display Page
/*
 String date = String(now.day()) + "/" + String(now.month()) + "/" + String(now.year());
 String times = String(now.hour()) + ":" + String(now.minute()) + ":" + String(now.second());
 myNextion.setComponentText("t009", date);
 myNextion.setComponentText("t010", times );  */
//kp = total_wind * humidity;
if ( enable == 0 )
{
 settime_begin -= 1;
 /*
   for (io = 0; io < 10; io++)
   {
     WATERValue  += hx711.read() / 1000;  }
   WATERValue  /= 10;
```

```
    WAter_test = map(WATERValue, 9530     , 10250  , 0    , 100) ;
    if ( WAter_test <= 0 ) WAter_test = 0;
    else if ( WAter_test >= 100 ) WAter_test = 100;
    Serial.print("Water Lavel: ");
    Serial.print(WAter_test / 100.0);
    Serial.print(" L");
    Serial.print("  ADC  ");
    Serial.println(WATERValue);
    Serial.print("\t\t");
    epan_new = (WATERValue - 14282) / 6.6069;
//************************************************************** แก้ไขสมการ จุดที่ 1
    //epan_new = 50;
    Serial.print(" epan_new = ");
    Serial.println(epan_new, 2);
  */
  int adc0 = analogRead(sensor);
  Serial.println("                                adc0 = " + String(adc0));
  epan_new = ((adc0 - 640.09) / 5.7089)*-1;
  Serial.println("                                                After  epan_new = " + String(epan_new,
2));
  myNextion.setComponentText("t103", String(epan_new, 1) );
  delay(100);
  if (settime_begin <= 0 )
  {
   settime_begin = 0;
  }
  myNextion.setComponentText("t856", String(settime_begin) + " s" );
 }
 myNextion.setComponentText("t002", String(kc, 2) );
 myNextion.setComponentText("t003", String(sm1, 1) );
 myNextion.setComponentText("t004", String(epan_old, 1) );
```

กรมวิชาการเกษตร

```
  digitalWrite(solenoid1, HIGH);
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, HIGH);
  dht.begin();
  myNextion.init();
  Wire.begin(); // Start the I2C
  RTC.begin();  // Init RTC
  // RTC.adjust(DateTime(__DATE__, __TIME__));  // Time and date is expanded to date and time on your computer at compiletime
*******************************************************************************
} // end setup
void loop()
{
  DateTime now = RTC.now();
  dater = String(now.day()) + "/" +  String(now.month()) + "/" +  String(now.year()) ;
  tter = String(now.hour()) + ":" +  String(now.minute()) + ":" +  String(now.second()) ;
  Serial.println(dater + " " + tter);
  /*
   float ten_thousand_increment = EEPROM.read(adr_increment_ten_thousand) * 10000.00;
   float one_thousand_increment = EEPROM.read(adr_increment_one_thousand) * 1000.00;
   float one_hundred_increment = EEPROM.read(adr_increment_one_hundred) * 100.00;
   float one_ten_increment = EEPROM.read(adr_increment_one_ten) * 10.00;
   float one_one_increment = EEPROM.read(adr_increment_one_one) * 1.00;
   float p1_increment = EEPROM.read(adr_increment_p1) * 0.10;
   float p2_increment = EEPROM.read(adr_increment_p2) * 0.01;
   increment = ten_thousand_increment + one_thousand_increment + one_hundred_increment + one_ten_increment + one_one_increment + p1_increment + p2_increment;
  */
  float hun_epo = EEPROM.read(adr_epo_hun) * 100.00;
  float ten_epo = EEPROM.read(adr_epo_ten) * 10.00;
  float one_epo =  EEPROM.read(adr_epo_one) * 1.00 ;
```

```
  float p1_epo = EEPROM.read(adr_epo_p1) * 0.10 ;
  float p2_epo = EEPROM.read(adr_epo_p2) * 0.01 ;
  epan_old = hun_epo + ten_epo + one_epo + p1_epo + p2_epo ;
  //-----------------------------------------------------------------
  float hun_flow = EEPROM.read(adr_flow_hun) * 100.00;
  float ten_flow = EEPROM.read(adr_flow_ten) * 10.00;
  float one_flow = EEPROM.read(adr_flow_one) * 1.00;
  flow_rate = hun_flow + ten_flow + one_flow;
  //-----------------------------------------------------------------
  //adr_tree_ten
  float ten_tree = EEPROM.read(adr_tree_ten) * 10.00;
  float one_tree = EEPROM.read(adr_tree_one) * 1.00;
  float p1_tree = EEPROM.read(adr_tree_p1) * 0.10;
  tree_radius = ten_tree + one_tree + p1_tree;
  kc = EEPROM.read(adr_kc) / 100.00;
  sm1 = EEPROM.read(adr_sm1) / 10.0;
  Serial.println("                    mem = " + String(EEPROM.read(adr_sm1_begin) / 10.0));
  Serial.println();
  Serial.println();
  //=============================================================
  //kc
  float one_kc = EEPROM.read(adr_kc_one) * 0.10;
  float p1_kc = EEPROM.read(adr_kc_p1) * 0.01;
  kc = 0.00 + one_kc + p1_kc;
  Serial.println("                    kc  mem = " + String(kc, 2));
  Serial.println();
  Serial.println();
  // tm = EEPROM.read(adr_tm);
  // sec = EEPROM.read(adr_sec);
  // mins = EEPROM.read(adr_mins);
  // starts = EEPROM.read(adr_starts);
```

กรมวิชาการเกษตร

```
String tr = "";
String sf = "";
String Epo = "";
String Epn = "";
int systems = 0;
int dutycycle = 1;
int starttime = 7;
float tm;
float kp;
float kc = 0.60;
float tree_radius ;
float flow_rate;
int checkbp = 1;
float bpp = 1.00;
float epan_new = 17.00;
float epan_old = 20.00;
float epan_cal = 0.00;
float sm1 = 1.6;
float humidity;
double total_wind = 0.00000;
#define pump 23
#define solenoid1 25
#define solenoid2 27
#define solenoid3 29
#define solenoid4 31
int rains = 7;
#define trig 3
int steps = 0;
int sec = 0;
int starts;
int enable = 0;
```

```
#include <SPI.h>
#include <SD.h>
File myFile; // สร้างออฟเจก File สำหรับจัดการข้อมูล
const int chipSelect = 53;
//---------------
#include <Wire.h>
#include <RTClib.h>
RTC_DS3231 RTC;
#define sensor A0
void setup() {
 Serial.begin(9600);
 pinMode(sensor, INPUT);
 pinMode(buzzer, OUTPUT);
 digitalWrite(buzzer, LOW);
 Serial.print("Initializing SD card...");
 pinMode(SS, OUTPUT);
 if (!SD.begin(chipSelect)) {
  Serial.println("initialization failed!");
  //return; }
 Serial.println("initialization done.");
 //myFile = SD.open("test2.txt", FILE_WRITE); /
 pinMode(rains, INPUT);
 Serial2.begin(9600);
 pinMode(trig, OUTPUT);
 digitalWrite(trig, HIGH);
 pinMode(pump, OUTPUT);
 pinMode(solenoid1, OUTPUT);
 pinMode(solenoid2, OUTPUT);
 pinMode(solenoid3, OUTPUT);
 pinMode(solenoid4, OUTPUT);
 digitalWrite(pump, HIGH);
```

กรมวิชาการเกษตร

```
#include <EEPROM.h>
int adr_epo_hun = 30;
int adr_epo_ten = 0;
int adr_epo_one = 1;
int adr_epo_p1 = 2;
int adr_epo_p2 = 3;
int adr_flow_hun = 4 ;
int adr_flow_ten = 5 ;
int adr_flow_one = 6 ;
int adr_tree_ten = 7;
int adr_tree_one = 8;
int adr_tree_p1 = 9;
int adr_kc_ten = 20;
int adr_kc_one = 21;
int adr_kc_p1 = 22;
int adr_kc = 10;
int adr_sm1 = 11;
int adr_sm1_begin = 12;
String dater = "";
String tter  = "";
String sme = "";
String Kkc;
//----------------------------------------
int adr_increment_ten_thousand = 13;
int adr_increment_one_thousand = 14;
int adr_increment_one_hundred = 15;
int adr_increment_one_ten = 16;
int adr_increment_one_one = 17;
int adr_increment_p1 = 18;
int adr_increment_p2 = 19;
int ss = 0;
```

```
int buzzer = 6;
float increment = 0.00;
int settime_begin = 30; //
//-------------------------------------------------------------
#include <Q2HX711.h>
const byte hx711_data_pin = 8;
const byte hx711_clock_pin = 9;
Q2HX711 hx711(hx711_data_pin, hx711_clock_pin);
float WATERValue = 0;
int io;
long WAter_test = 0;
//-------------------------------------------------------------
unsigned long previous = 0;
const long interval2 = 1000;
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
float mm;
#include <SoftwareSerial.h>
#include <Nextion.h>
SoftwareSerial nextion(10, 11);
Nextion myNextion(nextion, 9600);
#define epan A0
#define rain A1
float mins = 0.00;
float Etc;
float sm;
float T;
float Ei = 0.85;
int ts = 0;
```

ศูนย์วิทยทรัพยากร

```
String tr = "";
String sf = "";
String Epo = "";
String Epn = "";
int systems = 0;
int dutycycle = 1;
int starttime = 7;
float tm;
float kp;
float kc = 0.60;
float tree_radius ;
float flow_rate;
int checkbp = 1;
float bpp = 1.00;
float epan_new = 17.00;
float epan_old = 20.00;
float epan_cal = 0.00;
float sm1 = 1.6;
float humidity;
double total_wind = 0.00000;
#define pump 23
#define solenoid1 25
#define solenoid2 27
#define solenoid3 29
#define solenoid4 31
int rains = 7;
#define trig 3
int steps = 0;
int sec = 0;
int starts;
int enable = 0;
```

```
#include <SPI.h>
#include <SD.h>
File myFile; // สร้างออฟเจก File สำหรับจัดการข้อมูล
const int chipSelect = 53;
//---------------
#include <Wire.h>
#include <RTClib.h>
RTC_DS3231 RTC;
#define sensor A0
void setup() {
 Serial.begin(9600);
 pinMode(sensor, INPUT);
 pinMode(buzzer, OUTPUT);
 digitalWrite(buzzer, LOW);
 Serial.print("Initializing SD card...");
 pinMode(SS, OUTPUT);
 if (!SD.begin(chipSelect)) {
  Serial.println("initialization failed!");
  //return; }
 Serial.println("initialization done.");
 //myFile = SD.open("test2.txt", FILE_WRITE); /
 pinMode(rains, INPUT);
 Serial2.begin(9600);
 pinMode(trig, OUTPUT);
 digitalWrite(trig, HIGH);
 pinMode(pump, OUTPUT);
 pinMode(solenoid1, OUTPUT);
 pinMode(solenoid2, OUTPUT);
 pinMode(solenoid3, OUTPUT);
 pinMode(solenoid4, OUTPUT);
 digitalWrite(pump, HIGH);
```

กรมวิชาการเกษตร

```
  digitalWrite(solenoid1, HIGH);
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, HIGH);
  dht.begin();
  myNextion.init();
  Wire.begin(); // Start the I2C
  RTC.begin();  // Init RTC
  // RTC.adjust(DateTime(__DATE__, __TIME__));  // Time and date is expanded to date and time on your computer at compiletime
*******************************************************************************
} // end setup
void loop()
{
  DateTime now = RTC.now();
  dater = String(now.day()) + "/" +  String(now.month())  + "/" +  String(now.year()) ;
  tter = String(now.hour()) + ":" +  String(now.minute())  + ":" +  String(now.second()) ;
  Serial.println(dater + " " + tter);
  /*
   float ten_thousand_increment = EEPROM.read(adr_increment_ten_thousand) * 10000.00;
   float one_thousand_increment = EEPROM.read(adr_increment_one_thousand) * 1000.00;
   float one_hundred_increment = EEPROM.read(adr_increment_one_hundred) * 100.00;
   float one_ten_increment = EEPROM.read(adr_increment_one_ten) * 10.00;
   float one_one_increment = EEPROM.read(adr_increment_one_one) * 1.00;
   float p1_increment = EEPROM.read(adr_increment_p1) * 0.10;
   float p2_increment = EEPROM.read(adr_increment_p2) * 0.01;
   increment = ten_thousand_increment + one_thousand_increment + one_hundred_increment + one_ten_increment + one_one_increment + p1_increment + p2_increment;
  */
  float hun_epo = EEPROM.read(adr_epo_hun) * 100.00;
  float ten_epo = EEPROM.read(adr_epo_ten) * 10.00;
  float one_epo =  EEPROM.read(adr_epo_one) * 1.00 ;
```

```
  float p1_epo = EEPROM.read(adr_epo_p1) * 0.10 ;
  float p2_epo = EEPROM.read(adr_epo_p2) * 0.01 ;
  epan_old = hun_epo + ten_epo + one_epo + p1_epo + p2_epo ;
  //-------------------------------------------------------------
  float hun_flow = EEPROM.read(adr_flow_hun) * 100.00;
  float ten_flow = EEPROM.read(adr_flow_ten) * 10.00;
  float one_flow = EEPROM.read(adr_flow_one) * 1.00;
  flow_rate = hun_flow + ten_flow + one_flow;
  //-------------------------------------------------------------
  //adr_tree_ten
  float ten_tree = EEPROM.read(adr_tree_ten) * 10.00;
  float one_tree = EEPROM.read(adr_tree_one) * 1.00;
  float p1_tree = EEPROM.read(adr_tree_p1) * 0.10;
  tree_radius = ten_tree + one_tree + p1_tree;
  kc = EEPROM.read(adr_kc) / 100.00;
  sm1 = EEPROM.read(adr_sm1) / 10.0;
  Serial.println( "                    mem = "  + String(EEPROM.read(adr_sm1_begin) / 10.0));
  Serial.println();
  Serial.println();
  //============================================================
  //kc
  float one_kc = EEPROM.read(adr_kc_one) * 0.10;
  float p1_lc = EEPROM.read(adr_kc_p1) * 0.01;
  kc = 0.00 + one_kc + p1_lc;
  Serial.println( "                    kc  mem = "  + String(kc, 2));
  Serial.println();
  Serial.println();
  // tm = EEPROM.read(adr_tm);
  // sec = EEPROM.read(adr_sec);
  // mins = EEPROM.read(adr_mins);
  // starts = EEPROM.read(adr_starts);
```

ครบวงการปลาดอท

```
// steps = EEPROM.read(adr_steps);
// readsensor ();
/*
 if ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("recive : " + String(wind));
  total_wind = wind.toInt();
  Serial.println("                    total_wind  : " + String(total_wind));
  digitalWrite(buzzer, HIGH);
  delay(3000);
  starts = 1;
  delay(1000);
  digitalWrite(buzzer, LOW);
 }
*/
//========================================================= Display Page
/*
 String date = String(now.day()) + "/" + String(now.month()) + "/" + String(now.year());
 String times = String(now.hour()) + ":" + String(now.minute()) + ":" + String(now.second());
 myNextion.setComponentText("t009", date);
 myNextion.setComponentText("t010", times );  */
//kp = total_wind * humidity;
if ( enable == 0 )
{
 settime_begin -= 1;
 /*
   for (io = 0; io < 10; io++)
   {
    WATERValue  += hx711.read() / 1000;  }
   WATERValue  /= 10;
```

```
   WAter_test = map(WATERValue, 9530    , 10250  , 0    , 100) ;
   if ( WAter_test <= 0 ) WAter_test = 0;
   else if ( WAter_test >= 100 ) WAter_test = 100;
   Serial.print("Water Lavel: ");
   Serial.print(WAter_test / 100.0);
   Serial.print(" L");
   Serial.print("  ADC  ");
   Serial.println(WATERValue);
   Serial.print("\t\t");
   epan_new = (WATERValue - 14282) / 6.6069;
//********************************************************** แก้ไขสมการ จุดที่ 1
   //epan_new = 50;
   Serial.print(" epan_new = ");
   Serial.println(epan_new, 2);
  */
  int adc0 = analogRead(sensor);
  Serial.println("                                adc0 = " + String(adc0));
  epan_new = ((adc0 - 640.09) / 5.7089)*-1;
  Serial.println("                                        After  epan_new = " + String(epan_new,
2));
  myNextion.setComponentText("t103", String(epan_new, 1) );
  delay(100);
  if (settime_begin <= 0 )
  {
   settime_begin = 0;
  }
  myNextion.setComponentText("t856", String(settime_begin) + " s" );
 }
 myNextion.setComponentText("t002", String(kc, 2) );
 myNextion.setComponentText("t003", String(sm1, 1) );
 myNextion.setComponentText("t004", String(epan_old, 1) );
```

กรมวิชาการเกษตร

```
myNextion.setComponentText("t103", String(epan_new, 1) );
myNextion.setComponentText("t005", String(flow_rate, 1) );
myNextion.setComponentText("t006", String(total_wind) );
myNextion.setComponentText("t765", String(humidity) );
myNextion.setComponentText("t766", String(kp, 2) );
myNextion.setComponentText("t008", String(tree_radius, 1) );
int bstart = myNextion.getComponentValue("b0");
int benable = myNextion.getComponentValue("b1");
Serial.println("                bstart =  " + String(bstart));
Serial.println("                benable =  " + String(benable));
Serial.println();
if (settime_begin <= 0 )
{
 if ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("                recive : " + String(wind));
  int total_wind1 = wind.toInt();
  Serial.println("                total_wind  : " + String(total_wind1));
  total_wind = total_wind1 * 0.0005024 ;
  digitalWrite(buzzer, HIGH);
  delay(3000);
  starts = 1;
  delay(1000);
  digitalWrite(buzzer, LOW);
 }
 myNextion.setComponentText("t999", "Auto-Start");
 Serial.println("                เริ่มทำงาน !!!  ");
 humidity = dht.readHumidity();
 delay(2000);
 Serial.println("                humidity  :  " + String(humidity));
```

```
 Serial2.println('0');
 digitalWrite(trig, LOW); // Send to reset
 delay(1000);
 digitalWrite(trig, HIGH);
 delay(1000);
 settime_begin = 30;
 enable = 1;
}
if ( bstart == 1 )
{
 if ( Serial2.available ())
 {
  String wind = Serial2.readString();
  Serial.println("                start recive : " + String(wind));
  int total_wind1 = wind.toInt();
  Serial.println("                total_wind  : " + String(total_wind1));
  total_wind = total_wind1 * 0.0005024 ;
  Serial.println("                total_wind ok  : " + String(total_wind1, 6));
  digitalWrite(buzzer, HIGH);
  delay(3000);
  digitalWrite(buzzer, LOW);
 }
 starts = 1;
 beep ();
 enable = 1;
 myNextion.setComponentText("t999", "Start!!!");
 Serial.println("                เริ่มทำงาน !!!  ");
 humidity = dht.readHumidity();
 delay(2000);
 Serial.println("                humidity  :  " + String(humidity));
```

กรมวิชาการเกษตร

```
  //-------------------------------------------------------------------------------
  int adc0 = analogRead(sensor);
  Serial.println("                                adc0 = " + String(adc0));
  epan_new = ((adc0 - 640.09) / 5.7089)*-1;
  Serial.println("                                        After  epan_new = " + String(epan_new,
2));
  /*
    for (io = 0; io < 10; io++)
    {
     WATERValue  += hx711.read() / 1000;
    }
    WATERValue  /= 10;
    WAter_test = map(WATERValue, 9530    , 10250  , 0    , 100) ;
    if ( WAter_test <= 0 ) WAter_test = 0;
    else if ( WAter_test >= 100 ) WAter_test = 100;
    Serial.print("Water Lavel: ");
    Serial.print(WAter_test / 100.0);
    Serial.print(" L");
    Serial.print("  ADC  ");
    Serial.println(WATERValue);
    Serial.print("\t\t");
    // X= ระดับน้ำ
    // Y= ค่าที่วัดได้
    epan_new = (WATERValue - 14282) / 6.6069;
//************************************************************** แก้เวลแกาล จุดที่ 2
    //epan_new = 50;
  */
  Serial.println(epan_new, 2);
  myNextion.setComponentText("t103", String(epan_new, 1) );
  Serial2.println('0');
  delay(100); }
```

```
  if ( benable == 1 )
  {
   beep ();
   enable = 1;
   myNextion.setComponentText("t856", " " ); }
 //=============================================================================Set
KC

//=============================================================================
============================================== Set Epan Old
  int bkt0 = myNextion.getComponentValue("kt0");
  int bkt1 = myNextion.getComponentValue("kt1");
  int bkt2 = myNextion.getComponentValue("kt2");
  int bkt3 = myNextion.getComponentValue("kt3");
  int bkt4 = myNextion.getComponentValue("kt4");
  int bkt5 = myNextion.getComponentValue("kt5");
  int bkt6 = myNextion.getComponentValue("kt6");
  int bkt7 = myNextion.getComponentValue("kt7");
  int bkt8 = myNextion.getComponentValue("kt8");
  int bkt9 = myNextion.getComponentValue("kt9");
  int bkt_point = myNextion.getComponentValue("kt88");
  int bkt_del = myNextion.getComponentValue("kt99");
  if ( bkt0 == 1 )
  {
   Kkc += "0";
   kc = Kkc.toFloat();
   delay(1000);
   //-----------------------------------------
   myNextion.setComponentText("kt22", String(kc, 2) );
   beep();
  }
  if ( bkt1 == 1 )
```

สถาบันวิทยบริการ

```
{
 Kkc += "1";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep(); }
if ( bkt2 == 1 )
{
 Kkc += "2";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();  }
if ( bkt3 == 1 )
{
 Kkc += "3";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt4 == 1 )
{
 Kkc += "4";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt5 == 1 )
{
 Kkc += "5";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt6 == 1 )
{
 Kkc += "6";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt7 == 1 )
{
 Kkc += "7";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
 }
if ( bkt8 == 1 )
{
 Kkc += "8";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt9 == 1 )
```

กรมวิชาการเกษตร

```
{
 Kkc += "9";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt_point == 1 )
{
 Kkc += ".";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
if ( bkt_del == 1 )
{
 Kkc = "";
 kc = Kkc.toFloat();
 delay(1000);
 myNextion.setComponentText("kt22", String(kc, 2) );
 beep();
}
int kcs = kc * 100;
char datak[5];
String(kcs).toCharArray(datak, 5);
if ( kcs <= 99 )
{
 int one = datak[0] - 48;
 int p1 = datak[1] - 48;
 EEPROM.write(adr_kc_ten, 0);
 EEPROM.write(adr_kc_one, one);
 EEPROM.write(adr_kc_p1, p1);
 myNextion.setComponentText("kt22", String(kc, 2) );
}
/*
 if ( kcs >= 100 && kcs <= 999)
 {
 int ten = datak[0] - 48;
 int one = datak[1] - 48;
 int p1 = datak[2] - 48;
 EEPROM.write(adr_kc_ten, ten);
 EEPROM.write(adr_kc_one, one);
 EEPROM.write(adr_kc_p1, p1);
 myNextion.setComponentText("kt22", String(kc, 1) );
 }
*/
 //================================================
Tree Radius
 int bs0 = myNextion.getComponentValue("bs0");
 int bs1 = myNextion.getComponentValue("bs1");
 int bs2 = myNextion.getComponentValue("bs2");
 int bs3 = myNextion.getComponentValue("bs3");
 int bs4 = myNextion.getComponentValue("bs4");
 int bs5 = myNextion.getComponentValue("bs5");
 int bs6 = myNextion.getComponentValue("bs6");
 int bs7 = myNextion.getComponentValue("bs7");
 int bs8 = myNextion.getComponentValue("bs8");
 int bs9 = myNextion.getComponentValue("bs9");
 int bs_point = myNextion.getComponentValue("bs88");
 int bs_del = myNextion.getComponentValue("bs99");
```

กรมวิชาการเกษตร

```
if ( bs0 == 1 )
{
 tr += "0";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs1 == 1 )
{
 tr += "1";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs2 == 1 )
{
 tr += "2";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs3 == 1 )
{
 tr += "3";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs4 == 1 )
{
 tr += "4";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs5 == 1 )
{
 tr += "5";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs6 == 1 )
{
 tr += "6";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs7 == 1 )
{
 tr += "7";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs8 == 1 )
{
 tr += "8";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
```

กรมวิชาการเกษตร

```
}
if ( bs9 == 1 )
{
 tr += "9";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs_point == 1 )
{
 tr += ".";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
}
if ( bs_del == 1 )
{
 tr = "";
 tree_radius = tr.toFloat();
 myNextion.setComponentText("t014", String(tree_radius, 1) );
 beep();
 EEPROM.write(adr_tree_ten, 0);
 EEPROM.write(adr_tree_one, 0);
 EEPROM.write(adr_tree_p1, 0);
}
int trds = tree_radius * 10; // 9.5  , 45.4
char datac[5];
String(trds).toCharArray(datac, 5);
if ( trds <= 99 )
{
 int one = datac[0] - 48;
```

```
}
if ( bm9 == 1 )
{
 sf += "9";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm_point == 1 )
{
 sf += ".";
 flow_rate = sf.toFloat();
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
if ( bm_del == 1 )
{
 sf = "";
 flow_rate = sf.toFloat();
 EEPROM.write(adr_flow_hun, 0);
 EEPROM.write(adr_flow_ten, 0);
 EEPROM.write(adr_flow_one, 0);
 myNextion.setComponentText("t016", String(flow_rate, 1) );
 beep();
}
// if ( bm0 == 0 && bm1 == 0 && bm2 == 0 && bm3 == 0 && bm4 == 0 && bm5 == 0 && bm6 == 0
&& bm7 == 0 && bm8 == 0 && bm9 == 0)
// {
int flows = flow_rate; // 10-999
//Serial.println("                flows =  " + String(flows));
char datas[5];
```

กรมวิชาการเกษตร

```
  Serial2.println('0');
  digitalWrite(trig, LOW); // Send to reset
  delay(1000);
  digitalWrite(trig, HIGH);
  delay(1000);
  settime_begin = 30;
  enable = 1;
 }
 if ( bstart == 1 )
 {
  if  ( Serial2.available ())
  {
   String wind = Serial2.readString();
   Serial.println("                    start recive : " + String(wind));
   int total_wind1 = wind.toInt();
   Serial.println("                    total_wind  : " + String(total_wind1));
   total_wind = total_wind1 * 0.0005024 ;
   Serial.println("                    total_wind ok  : " + String(total_wind1, 6));
   digitalWrite(buzzer, HIGH);
   delay(3000);
   digitalWrite(buzzer, LOW);
  }
  starts = 1;
  beep ();
  enable = 1;
  myNextion.setComponentText("t999", "Start!!!");
  Serial.println("                    เริ่มทำงาน !!!  ");
  humidity = dht.readHumidity();
  delay(2000);
  Serial.println("                    humidity  :  " + String(humidity));
```

```
   mins += 1;
   sec = 0;
   //    EEPROM.write(adr_sec, sec);
   //   EEPROM.write(adr_mins, mins);
  }
  // String ttt = "Time = " + String(mins) + ":" + String(sec) + "  ";
  String ttt = "UT=" + String(tm, 2) + ",CT=" + String(mins) + ":" + String(sec) + "  ";
  myNextion.setComponentText("t999", ttt);
  //  Serial.println("              min4 = " + String(mins));
  //   Serial.println("              sec4 = " + String(sec));
  //    Serial.println("                  Value 4 - ON ");
  digitalWrite(solenoid1, HIGH); // value 1 - ON
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, LOW);
 }
 if ( (mins + sec / 100.00) * 60.00  >= ts && steps == 3 )
 {
  myNextion.setComponentText("t999", "Finish4");
  Serial.println("                  Value 4 - OFF ");
  digitalWrite(solenoid1, HIGH);
  digitalWrite(solenoid2, HIGH);
  digitalWrite(solenoid3, HIGH);
  digitalWrite(solenoid4, HIGH);
  delay(1000);
  Serial.println("                Pump - OFF ");
  digitalWrite(pump, HIGH); // pump OFF
  delay(3000);
  epan_old = epan_new;
  int eepo = epan_old * 100;
```

กรมวิชาการเกษตร

```
char data[6];
String(eepo).toCharArray(data, 6);
//---------------------------------------------
if ( eepo <= 999 )
{
 int one = data[0] - 48;
 int p1 = data[1] - 48;
 int p2 = data[2] - 48;
 EEPROM.write(adr_epo_hun, 0);
 EEPROM.write(adr_epo_ten, 0);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
if (  eepo >= 1000 && eepo <= 9999 )
{
 int ten = data[0] - 48;
 int one = data[1] - 48;
 int p1 = data[2] - 48;
 int p2 = data[3] - 48;
 EEPROM.write(adr_epo_hun, 0);
 EEPROM.write(adr_epo_ten, ten);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
//-----------------------------------------------------
if (  eepo >= 10000 && eepo <= 99999 )
{
```

```
 int hun = data[0] - 48;
 int ten = data[1] - 48;
 int one = data[2] - 48;
 int p1 = data[3] - 48;
 int p2 = data[4] - 48;
 EEPROM.write(adr_epo_hun, hun);
 EEPROM.write(adr_epo_ten, ten);
 EEPROM.write(adr_epo_one, one);
 EEPROM.write(adr_epo_p1, p1);
 EEPROM.write(adr_epo_p2, p2);
 myNextion.setComponentText("t100", String(epan_old, 1) );
}
// delay(1000);
sec = 0;
steps = 0;
starts = 0;
// EEPROM.write(adr_sec, sec);
// EEPROM.write(adr_starts, starts);
// EEPROM.write(adr_steps, steps);
 }
 }
 } // end millis
}// end loop
void beep ()
{
 digitalWrite(buzzer, HIGH);
 delay(80);
 digitalWrite(buzzer, LOW);
 delay(80);
}
```

กรมวิชาการเกษตร